

Oracle® Cloud

Using Integrations in Oracle Integration 3



F45537-88
January 2025



Oracle Cloud Using Integrations in Oracle Integration 3,

F45537-88

Copyright © 2022, 2025, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

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

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

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

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

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Audience	xv
Documentation Accessibility	xv
Diversity and Inclusion	xv
Related Resources	xvi
Conventions	xvi

1 Learn About Integrations

10 Things to Know Before You Start	1-1
Watch a Video	1-4
Integration Elements	1-4
Integration Patterns	1-5
Integration Adapters	1-5
Workflows for Integrations	1-5
Prerequisites for Developing Integrations	1-5
Workflow for Developing an Integration	1-7
Workflow for Testing Integrations	1-9
Workflow for Deploying Automation Solutions	1-12
Workflow for Monitoring Integrations	1-13
Initial Monitoring Tasks	1-13
Daily Monitoring Tasks	1-14
Occasional Monitoring Tasks	1-15
Tours and Tutorials	1-16
Explore the Navigation Pane	1-17
Tour: Explore an Integration That Sends a Hello World Message	1-21
Tour: Explore a Multiple Verbs and Resources Invoke Integration	1-23
Tutorial: Build Your First Integration From Scratch	1-27
Tutorial: Build Your Second Integration From Scratch	1-53
Get Started with Projects	1-80
About Projects	1-80
Project Guidance If You've Used Packages	1-82
Tour of the Projects Workspace	1-83
Navigation to the Projects Workspace	1-83

Design Tab	1-84
Deploy Tab	1-87
Observe Tab	1-87
10 Tips for Working with Projects	1-89
4 Ways to Organize Projects	1-91
About Project Deployment	1-92
Restrictions for Projects	1-93
Understand Integration Patterns	1-93
About Application Integrations	1-95
How to Choose Between Synchronous and Asynchronous Integrations	1-95
How to Specify if an Integration is Synchronous or Asynchronous	1-98
Differences Between Synchronous and Asynchronous Integrations	1-99
About Schedule Integrations	1-99
About Event Integrations	1-100
Understand Hybrid Integrations	1-100
About Connecting to On-Premises Applications	1-101
About Creating Hybrid Integrations	1-101
About the Connectivity Agent	1-102
Connection Patterns for Hybrid Integrations	1-105
About Connecting to Private Resources	1-109
Outbound Private Connection Pattern	1-109
Adapters that Support Connecting to Private Endpoints	1-110
Requirements for Connecting to a Private Endpoint	1-111
Integration Concepts	1-111
About Integrations	1-112
About Connections	1-113
Available Adapters for Connections	1-115
Understand Trigger and Invoke Connections	1-119
Trigger Connections	1-120
Invoke Connections	1-122
About Mappings	1-126
About Lookups	1-129
About Packages	1-130
About Business Identifiers for Tracking Fields in Messages	1-131
About Error Management	1-132
Get Started with Observability	1-133
About the Observability Pages	1-134
Develop an Observability Strategy	1-136
About Building Your Own Observability Tools	1-139
About Oracle Integration	1-139
Start Oracle Integration	1-139

2 Best Practices

Avoid Common Integration Pattern Pitfalls	2-1
Design Best Practices	2-5
Create Schedule Integrations Mindfully	2-5
Follow Guidelines for Synchronous Integrations	2-6
Design Long-Running Integrations as Asynchronous	2-7
Understand Timeouts in Synchronous Integrations	2-7
Send Data to Third-Party Systems Efficiently	2-7

3 Design and Monitor Integrations in Projects

Design a Project	3-1
Control Who Can Edit, View, and Monitor in a Project	3-2
Create or Import a Project	3-14
Create, Import, or Copy an Integration into a Project	3-19
Unlock an Integration in a Project	3-21
Refresh Endpoints of an Integration in a Project	3-21
Update Property Values of an Integration in a Project	3-21
Create a Version of an Integration in a Project	3-22
Add and Share a Connection Across a Project	3-22
Edit a Connection in a Project	3-30
Refresh Connection Metadata in a Project	3-31
Delete a Connection in a Project	3-31
Unlock a Connection in a Project	3-31
Add a Lookup to a Project	3-31
Edit a Lookup in a Project	3-32
Clone a Lookup in a Project	3-32
Delete a Lookup in a Project	3-33
Import a JavaScript Library in a Project	3-33
Delete a JavaScript Library in a Project	3-33
Update a JavaScript Library in a Project	3-34
Create, Publish, and Subscribe to Events in a Project	3-34
Create Robots in a Project	3-35
Create Healthcare Integrations in a Project	3-36
Create B2B Integrations in a Project	3-36
Manage a Project	3-37
Create and Manage a Project Deployment	3-38
Understand the Integration Versioning Life Cycle in a Project Deployment	3-38
Create a Deployment	3-39

Manage Project Deployments	3-40
Activate or Deactivate a Project	3-41
Edit and Replace Dependent Resources in a Project	3-42
Export a Project	3-44
Clone a Project	3-45
Invoke Child Integrations Inside or Outside of Projects	3-46
Manage Accelerator Projects	3-50
Extend an Integration in an Accelerator Project	3-50
Upgrade an Accelerator Project and Merge Extensions	3-66
Activate or Deactivate an Integration in a Project	3-73
Deploy Integration Endpoints to Oracle Cloud Infrastructure API Gateway	3-74
Clone an Integration in a Project	3-84
Update the Tracing Level of Integrations in a Project	3-85
Test REST Adapter Trigger Connection-Based Integrations in Projects	3-85
View the Dependent Relationships Between Project Resources	3-85
View Run Details About Integrations in Projects	3-93
View Project Status	3-93
Edit Project Description Details	3-94
Delete a Project	3-94
Monitor Integrations in a Project	3-94
Monitor the Message Processing Status of Integrations in Projects	3-94
Track Instances in a Project	3-96
View Subscribed Instances in a Project	3-97
Manage Errors in a Project	3-98
View the Schedule Integration Runs Calendar in a Project	3-99
View the Design-Time Audit Log of a Project	3-99
Monitor Robots in a Project	3-100

4 Create Connections

Inbound and Outbound Endpoints and Transport Layer Security Server Version Support	4-1
Create a Connection	4-1
Configure Connection Properties and Security Properties	4-3
Configure Connection Properties	4-3
Configure Connection Security	4-7
Test the Connection	4-11
Edit a Connection	4-12
Clone a Connection	4-12
Delete a Connection	4-13
Unlock a Connection	4-13
Refresh Integration Metadata	4-13
Manage Security Certificates	4-14

Upload a Certificate to Connect with External Services	4-14
Update or Delete a Certificate	4-17
View Preinstalled Adapters	4-17
Allow JSON Numbers with High Precision and Scale	4-18

5 Create Application Integrations

Create an Integration	5-1
Design an Application Integration	5-3
Define Inbound Triggers and Outbound Invokes	5-6
Add Suggestions, Invokes, and Actions Through an Inline Menu	5-9
Cut and Paste Elements in the Integration Canvas	5-11
Expand and Collapse Actions in the Integration Canvas	5-16
Create Global Variables	5-17
Define Ad-Hoc Maps	5-18
Import a Map File into an Integration	5-20
Add Actions to an Application Integration	5-20
Loop over Repeating Elements with a For-Each Action	5-21
Route Expressions with Switch Action Branches	5-24
Manage a Group of Actions and Fault Handlers with a Scope Action	5-27
Assign Values to Scalar Variables in an Assign Action	5-31
Loop Over Actions or Invoke Connections While a Condition is Satisfied with a While Action	5-34
Send Notification Emails During Stages of the Integration with a Notification Action	5-36
Set the content-type in a Notification Action	5-40
Build Complex Assignment Statements with a Stitch Action	5-40
Delay Integration Processing for a Specified Time Period with a Wait Action	5-44
Add Global Fault Handling to Integrations	5-47
Catch Faults with a Re-throw Fault Action	5-49
Throw Faults with a Throw New Fault Action	5-50
Process Files in Schedule Integrations with a Stage File Action	5-53
Configure a Stage File Action	5-55
Receive Requests for Multiple Resources in a Single REST Adapter Trigger with a Pick Action	5-66
Log Messages with a Logger Action	5-69
Process Tasks in Parallel with a Parallel Action	5-70
Invoke Oracle Cloud Infrastructure Functions Directly from an Integration with an OCI Function Action	5-77
Publish Events in an Integration with a Publish Event Action	5-85
Add JavaScript Functions to an Integration with a JavaScript Action	5-86
Invoke Oracle Cloud Infrastructure Object Storage from an Integration with an OCI Object Storage Action	5-88
Add Placeholder Notes with a Note Action	5-93

Translate an EDI Document with the B2B Action	5-94
Convert HL7 Messages with a Healthcare Action	5-96
Interact with Files in File Server	5-100
Call a Robot from an Integration	5-105
Use XPath Axis and Wildcard Expressions in the Expression Builder	5-106
Use Lookups in Variable Assignments	5-107
Define Fault Aggregation in Parallel Processing Scenarios	5-110
Assign Business Identifiers for Tracking Fields in Messages	5-112
Assign Business Identifiers	5-112
Delete Business Identifiers	5-113
Maintain Business Identifier Definitions During Trigger Connection/Schedule Integration Parameter Changes	5-113
Display Errors and Warnings in an Integration	5-114

6 Create Schedule Integrations

Create a Schedule Integration	6-1
Define the Integration Schedule	6-2
Create and Update Parameters in Schedule Integrations	6-6
Override Schedule Parameter Values	6-9
Run a Schedule Integration Immediately	6-10
Start and Pause an Integration Schedule	6-11
Change the User Submitting an Integration Schedule	6-14
Globally Change the Submitter of Integration Schedules	6-15
View Past and Present Schedule Integration Runs	6-15
View Future Schedule Integration Runs	6-15
Edit an Integration Schedule	6-16
Delete an Integration Schedule	6-16
Unlock an Integration Schedule	6-17
Convert a Schedule Integration to a REST Adapter-Triggered, Application Integration	6-17
Delete Scheduled or Ad-Hoc Runs	6-18
Monitor Schedule Integration Runs	6-19
Understand Schedule Integration Behavior When Using the Local Time Zone	6-19

7 Create Event Integrations

About Publishing and Subscribing to Events	7-1
Create Events to Publish and Subscribe to in Integrations	7-2
Publish Events in an Integration	7-4
Subscribe to Events in an Integration	7-6
Retain an Event Subscription During Integration Deactivation	7-11
Define Custom Header Filters to Trigger Integrations	7-12

8 Map Data and Create Lookups

Map Data	8-1
Access the Mapper	8-1
Encode and Decode File Attachment Content	8-1
Import Map Files	8-3
Import a Map File into Oracle JDeveloper	8-3
Import a Map File into Oracle Integration	8-5
Manage Lookups	8-5
Create a Lookup	8-6
Add Adapters or Domain Names to a Lookup	8-6
Clone a Lookup	8-7
Delete a Lookup	8-8
Update Lookups in Active Integrations	8-8
Export a Lookup	8-8
Import a Lookup	8-8
Create the lookupValue Function	8-9
Access the Build Lookup Function Wizard	8-9
Select the Lookup	8-12
Select the Source and Target Columns	8-12
Specify the Default Value	8-13
Review Your Lookup Table Selections	8-13

9 Manage the Agent Group and Connectivity Agent

Workflow for Using the Connectivity Agent	9-1
Requirements for the Connectivity Agent	9-2
Key Points for the Installation of the Connectivity Agent	9-2
Download and Run the Connectivity Agent Installer	9-3
Create an Agent Group	9-3
System Requirements	9-5
Connectivity Agent Restrictions	9-7
Agent Download and Installation	9-8
Download and Install the Agent	9-8
Install a Certificate on the Agent Host	9-10
Run the Connectivity Agent Installer as a Background Process on Linux Systems	9-10
Restart the Connectivity Agent	9-11
Create a Connection with an Agent Group	9-12
Upgrade the Connectivity Agent Automatically	9-13
Recover the Connectivity Agent in Case of Host Failure	9-14

Use the Connectivity Agent in High Availability Environments	9-15
Delete an Agent Group	9-16
Agent Behavior in a Decommissioned Instance or HTTP 404/401 Error Response Codes	9-17

10 Use JavaScript Libraries in Integrations

Use Libraries to Manage Integration Functions	10-1
View Library Details	10-1
Edit Library Function Values	10-2
Determine Which Integrations Use a Library Function	10-2
Delete a Library	10-3
Import a Library File	10-3
Write JavaScript	10-3
Export a Library File	10-6
Update JavaScript Files	10-6
Use Library Functions in Integrations	10-9

11 Manage Integrations

Activate and Deactivate Integrations	11-1
Activate an Integration	11-2
Reactivate Integrations after a Connection Update	11-4
Deactivate an Integration	11-5
Deactivate an Integration with Business Events	11-6
Manage Tracing Levels on Integrations	11-7
Test Integrations in Oracle Integration	11-9
Test Integrations from Inside the Integration Canvas	11-9
Test Integrations from Outside the Integration Canvas	11-13
Operation Section	11-13
Request and Response Sections	11-15
Edit and Replace Dependent Resources	11-17
Edit and Replace Dependent Resources in a Package	11-17
Edit and Replace Dependent Resources in an Integration	11-20
Unlock Dependent Resources	11-22
Invoke a Child Integration from a Parent Integration	11-23
Create a Co-located Integration with Header Support	11-27
Backward Compatibility for Header Support	11-30
Dynamically Invoke a Child Integration	11-30
Manage Packages	11-33
View the Integrations in a Package	11-33
Import a Package	11-34
Export a Package	11-35

Update the Package for an Integration	11-35
Delete a Package	11-35
Convert a Package to a Project	11-36
Import and Export Integrations	11-36
Import an Integration	11-37
Import a Prebuilt Integration	11-37
Export an Integration	11-38
Clone the Design-Time Metadata of an Entire Service Instance	11-38
Step 1: Create an Object Storage Bucket and Construct the Storage URL	11-38
Step 2: Configure the Instance Object Storage Bucket	11-42
Step 3: Export and Import Design-Time Metadata Between Instances	11-43
Create an Export Job	11-43
Create an Import Job	11-45
Refresh Endpoints for Integrations	11-48
Set the Time Limit for Inactive Sessions	11-50
Modify an Integration	11-51
Impact on the Mapper of Editing the Endpoint Information in an Integration	11-51
View the Actions and Connections in an Integration	11-52
Clone an Integration	11-53
Create a Draft of an Integration	11-54
Delete an Integration	11-54
Recover Unsaved Integration Changes	11-54
Override Design-Time Properties in an Integration	11-56
Use Metadata in Integrations	11-58
Cancel a Running Integration Instance	11-60
View the Contents of an Incomplete Integration	11-60
Edit an Integration with Incomplete Connections	11-61
Change the Time Zone	11-61

12 Monitor Integrations During Runtime

View the Dashboard	12-1
Monitor the Number of Emails Sent from Oracle Integration	12-5
Check the Audit History for an Integration or Other Component	12-5
See the Current Status of All Integration Components	12-8
Check an Integration's Tracing Level	12-9
View Notification Alert Announcements and the Status of Asynchronous Operations	12-10
View Notification Alert Announcements in the Banner	12-10
View the Progress of Asynchronous, Design-Time Operations	12-11
Monitor the Message Processing Status of Integrations	12-12
Monitor Agents	12-17
Learn About the Payloads in the Activity Stream	12-18

Measure an Integration's Performance	12-20
Determine an Integration's Typical Processing Times	12-21
Determine an Integration's Performance on a Given Day	12-22
Determine Whether an Integration's Performance Is Consistent Over Time	12-22
Find Integration Instances with Increased Processing Times	12-23
Measure the Running Times of an Integration's Actions	12-24
Find Integrations That Experienced Errors	12-24
Find Integrations with Errors	12-25
Determine the Source of an Integration's Error	12-25
Find the Integration Error Rate for the Last 24 Hours	12-27
Measure Your Integration Errors Over Time	12-27
Find the Connections with the Most Errors	12-28
Find Integrations That Timed Out or Were Aborted	12-30
Manage Errors	12-30
View Error Messages and the Message Payload in the Activity Stream of Failed Instances	12-31
Resubmit Failed Messages	12-39
Discard Errors	12-42
View the Status of Message Recovery	12-43
View Errors By Integration and Connection	12-44
View the Business Identifiers Used in Failed Integration Instances	12-45
Track Integration Instances	12-46
Filter the Display of Integration Instances	12-46
Search for Business Identifier Values	12-49
Track the Status of Integration Instances	12-54
Track the Integration Flow on the Instance Details Page	12-68
Determine Why an Integration Instance Stopped Running	12-72
Replay Integration Instances	12-74
Monitor Integrations That Subscribe to Events	12-76
View Minimal Details About Debug Tracing Level Instances for 32 Days	12-77
View the Calendar of Schedule Integration Runs	12-78
Diagnose and Manage Event-Based Oracle Fusion Applications Integrations	12-79
Monitor Business Event Delivery Status	12-81
Send System Status Reports with Notification Emails	12-83
Understand DKIM Authentication and SPF Record Details	12-84
Configure Notification Emails	12-84
Remove Email Addresses from the Suppression List	12-88
Change the Delivery Time of Notification Emails	12-88

13 Troubleshoot Oracle Integration

Troubleshoot Oracle Cloud Infrastructure Notification Email Configuration to Ensure Proper Delivery	13-1
Troubleshoot the Connectivity Agent	13-2
Troubleshoot Private Endpoints	13-11
Troubleshoot Integration Runtime	13-13
An Integration Runs Slowly	13-14
Fewer Integration Instances Ran Than Usual	13-15
Using Basic Authentication to Call Services When Oracle Identity Cloud Service is Configured With Multi-Factor Authentication Causes a 401-Unauthorized Error	13-16
Perform Remedial Actions When the Payload Exceeds the Size Limit	13-16
Remove Unprintable Control Characters from the Incoming Payload	13-17
Notification Action Returns a <from> value is invalid Error	13-18
Synchronous Integration Timeout Failure	13-19
Activity Stream Details	13-19
Message Dequeuing Takes Three to Four Seconds in Asynchronous Integrations	13-20
403 Error When Accessing an API Protected Using OAuth	13-20
Troubleshoot Certificate Import Failures	13-20
Troubleshoot Integration Design Time	13-21
Missing a Primary Business Identifier in an Integration	13-21
Unable to Call a replaceAll JavaScript Function	13-21
Conflict with Using Variables of String Data Type and Applying Mathematical Comparator Expressions	13-22
Reduce Network Calls that Slow Down Processes Caused By Writing Record By Record	13-22
Invoke Subprocess Integrations Using a Single REST API Connection	13-22
Trigger Connection Cannot Receive an Empty Array in JSON if the Corresponding JSON Received Has an Empty JSON	13-23
Troubleshoot Schedule Integrations	13-23
Schedule Integrations Are Not Running on Time	13-24
When a Schedule Integration Instance Gets Terminated	13-24
Troubleshoot Integration Activations	13-24
Troubleshoot Integration Activations	13-25
Delay in Message Processing of First Instance after Activation	13-26

A Oracle Integration Use Cases

Manage Oracle Cloud Infrastructure Buckets and Objects from an Integration	A-1
Read Large XML Files Containing Multiple Namespaces	A-8
Use a Stage File Write Operation with an Opaque Schema	A-11
Test REST Adapter Trigger-Based Integrations with Multipart Attachments on the Configure and run page	A-12

Preface

This guide describes how to use Oracle Integration to integrate your applications.

 **Note:**

The use of this adapter may differ depending on the features you have, or whether your instance was provisioned using Standard or Enterprise edition. These differences are noted throughout this guide.

Topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Diversity and Inclusion](#)
- [Related Resources](#)
- [Conventions](#)

Audience

This guide is intended for users who want to create, activate, and monitor application integrations.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <https://www.oracle.com/corporate/accessibility/>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <https://support.oracle.com/portal/> or visit [Oracle Accessibility Learning and Support](#) if you are hearing impaired.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve.

Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Resources

See these Oracle resources:

- Oracle Cloud at <http://cloud.oracle.com>
- *Using the Oracle Mapper with Oracle Integration 3*

Conventions

The following text conventions are used in this document:

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

Learn About Integrations

Become an expert at integrations by learning about their components, completing product tours and tutorials, and reviewing tips.

Topics:

- [10 Things to Know Before You Start](#)
- [Watch a Video](#)
- [Workflows for Integrations](#)
- [Tours and Tutorials](#)
- [Get Started with Projects](#)
- [Understand Integration Patterns](#)
- [Understand Hybrid Integrations](#)
- [Integration Concepts](#)
- [Get Started with Observability](#)
- [About Oracle Integration](#)

10 Things to Know Before You Start

New to Oracle Integration? Whether you're a seasoned developer or have done a little work in low-code platforms, these pointers will help you get started with confidence and become an expert quickly.

Oracle Integration has an intuitive drag-and-drop interface that lets you design integrations quickly and easily. To ensure that you're building efficient integrations, spend some time developing your expertise before you start building.

1. Integrations Streamline Your Business

No matter your business or your goals, you can use Oracle Integration to automate your business processes. For example, you can:

- Send data between your applications.
- Connect applications in the cloud with on-premises software.
- Automate end-to-end solutions, such as procure to pay, inquiry to order, and hire to retire.
- Monitor your business processes.
- Recover from errors.

Several components help move data, including connections, mappings, and lookups. Understand how they work together so you can grow your skills and build integrations quickly and efficiently.

Learn more: [About Integrations](#), [About Connections](#), [About Mappings](#), and [About Lookups](#)

2. A Project Is Your Super-Powered Release Manager

Oracle Integration offers an intuitive user interface that places all the components that you need in a single solution, called a project. There, you design, manage, and observe related integrations. A project streamlines your day-to-day work and your integration deployment.

Learn more: [About Projects](#)

3. An Integration's Pattern Determines How it Starts

When you begin designing an integration, you choose the integration pattern to use. The pattern determines the event or activity that starts the integration.

The following integration patterns are available:

- **Application integrations**

Create an application integration to move data after a specific business event occurs or after a change in an application occurs. For example, an application integration runs after a customer submits an order, or after an employee makes an update in an application.

When the business event occurs, the application sends a request to Oracle Integration to start the integration. A trigger connection handles the request. You create the trigger connection using an adapter.

- **Event integrations**

Create an event integration to move data when someone creates a specific event type in an application. This publish-subscribe pattern requires two integrations. The first integration detects that a specific event type has been created in an application and publishes the event. The second integration subscribes to the event.

- **Schedule integrations**

Create a schedule integration to move data, either at a specific time and date or on demand.

Learn more: [Understand Integration Patterns](#)

4. Application Integrations Can Be Asynchronous or Synchronous

You can design an application integration as either a synchronous or asynchronous integration. Synchronous integrations deliver immediate responses, whereas asynchronous integrations guarantee the delivery of requests, no matter how complex the integration is.

- **Asynchronous integrations are guaranteed to run when they are accepted**

Use asynchronous integrations whenever you can. An asynchronous integration provides an immediate acknowledgment and, depending on the design, it can also provide a response upon completion.

- **Synchronous integrations give immediate responses**

When your business process requires an immediate response, such as providing a customer with a confirmation number for an order, build a synchronous integration. A synchronous integration finishes quickly, provides a response back to the caller, and raises any errors back to the caller.

Errors that occur for synchronous integrations are unrecoverable. For example, if an error occurs, you can't submit the failed integration instance.

See [About Application Integrations](#).

5. Schedule Integrations Should be Staggered and Simple

When your business response must run on a schedule, use a schedule integration. For example, use a schedule integration to run payroll at the end of the month or to synchronize your inventory every Friday at 5 PM.

Keep in mind the following guidance when creating schedule integrations:

- Stagger the start times.

For example, if you need to run ten daily integrations at the end of the business day, start one at 5 PM, the next at 5:05 PM, and so on.

- Keep schedule integrations small and simple.

Keep reading to learn tips for keeping schedule integrations small and simple.

See [About Schedule Integrations](#).

6. You Can Combine Integrations with Different Patterns

If you need an immediate response or need an integration to run on a schedule, you can pair a synchronous or schedule integration with an asynchronous integration.

- **When to pair a synchronous and asynchronous integration**

When a business process is complex and requires an immediate response, build a synchronous integration that completes two tasks:

1. Returns a quick response.

For example, a synchronous integration can return the confirmation number for a customer's order or complete a credit check for a customer before fulfilling an order.

2. Kicks off one or more asynchronous integrations.

The asynchronous integrations contain the bulk of the work, such as updating inventory, creating packing lists, and more.

See [How to Choose Between Synchronous and Asynchronous Integrations](#).

- **When to pair a schedule and asynchronous integration**

If a business process is complex, build a schedule integration that completes two tasks:

1. Starts the process at the right time.

2. Kicks off one or more asynchronous integrations, which contain the bulk of the work.



Tip:

No matter what type of integration you create, try to keep it as simple as possible. Simple integrations run more quickly than complex integrations. Of course, an integration can be only as simple as its corresponding business process, and some business processes are overwhelmingly complex. When designing for such complexity, remember that your integrations are competing for resources. Make sure that they all get a fair chance to complete as planned.

7. Service Limits Provide Guard Rails

Everyone must work within the service limits, which are the quotas and allowances for working with integrations. A common mistake that new users make is designing a complex integration that exceeds the service limits.

Your focus right now is understandably on getting your business flow started, but spend some time thinking about the scenarios the integration will encounter in the real world. For example, before you start designing a synchronous integration that needs to process orders, consider the real-world size of the files that the integration must process. For example, if the order that the integration must process contains many lines, the order processing could take a long time, and the integration might time out.

You don't need to memorize the service limits. However, reference them often when planning and developing an integration. Know the limits up front so you can design effective and appropriate integrations from the beginning.

Learn more: Service Limits in *Provisioning and Administering Oracle Integration 3*

8. Best Practices Are Available

You don't have to be an expert at Oracle Integration to create best-in-class integrations. That's because we collected the best practices and antipatterns, also known as worst practices. Follow this guidance so you know the actions to take—and the mistakes to avoid. Review this guidance before and while you work so that you create integrations that go live without issues.

Knowing the best practices and antipatterns, along with the service limits, will make you the go-to resource for integration best practices at your organization.

Learn more: [Design Best Practices](#)

9. Fault Handlers Keep Things Moving When Mistakes Happen

Before you start designing your first integration, spend some time implementing a global fault handler. A fault handler translates cryptic error messages into actionable tasks. This small time investment keeps your business running smoothly and saves everyone time.

10. Test Now, or Rework After Go-Live

An integration that performs well in isolation might perform well under load...or it might not. You won't know how all your systems work together until you test them in a production-like environment. For instance, an on-premises application that's experiencing some latency could negatively impact your synchronous integration.

Before you move your integrations to a production environment, test them in a real-world scenario, under load. If you need to modify the integration, it's much faster and easier to rework it before you start using it in a production environment.

Watch a Video

Watch a video that introduces key concepts about integrations.

Integration Elements

Learn the basic vocabulary and concepts used in Oracle Integration.



[Video](#)

Integration Patterns

Take a look at the various integration styles, or patterns, available in Oracle Integration, and how, why, and when you might use them.



[Video](#)

Integration Adapters

Get a high level look at adapters in Oracle Integration. Learn what they do, and what kinds are available for your integrations.



[Video](#)

Workflows for Integrations

Complete several prerequisite tasks, and then you can start designing and developing an integration. Next, it's time to deploy to a higher environment and then test and monitor your integration.

Topics:

- [Prerequisites for Developing Integrations](#)
- [Workflow for Developing an Integration](#)
- [Workflow for Deploying Automation Solutions](#)
- [Workflow for Testing Integrations](#)
- [Workflow for Monitoring Integrations](#)

Prerequisites for Developing Integrations

Depending on your organization's requirements, you might need to perform some, all, or none of the prerequisites for designing integrations.

Complete the required prerequisites one time, before you start designing integrations.

Step	Task	More information
1	Plan your work	<p>Before you start building, spend some time planning your work. For example:</p> <p class="list-item-l1">1. Understand the problem.</p> <p>An integration automates work that people would otherwise complete. Spend some time understanding the reason the work needs to be done, the current process, the problems with the current process, and the ideal workflow. Gathering requirements can be challenging, but this research becomes a worthwhile use of your time when you collect information that helps you deliver a superb solution.</p> <p>For example, you might need to:</p> <ul style="list-style-type: none">• Create an order in Oracle ERP Cloud when Oracle ERP Cloud receives an opportunity won event from a CRM.• Synchronize patient data every Friday at 5 PM. <p class="list-item-l1">2. Identify how to fulfill the requirements in an integration and plan your integration at a high level.</p> <p class="list-item-l1">3. Identify the applications that you're connecting, and collect connectivity information.</p> <p>For example, collect the URLs and authentication details for the applications.</p> <p class="list-item-l1">4. Determine whether a recipe or accelerator can help.</p> <p>Recipes and accelerators contain all the resources required for specific integration scenarios. See Recipes on the Oracle Help Center.</p>
2	Complete the prerequisites for the applications you're connecting	<p>Required if: The adapter you're using to connect to an application has prerequisites. Most adapters have prerequisites.</p> <p>Get started: Complete the prerequisites described in the guides for each adapter. Additionally, determine the security policies and connection login credentials that you'll use for the connections that are based on the adapter. See Configure Connection Properties.</p>

Step	Task	More information
3	Install the on-premises connectivity agent	<p>Required if: An integration needs to connect to applications that reside in an on-premises network.</p> <p>Not needed if: You're connecting to cloud applications only.</p> <p>Learn more: About Connecting to On-Premises Applications.</p> <p>Get started:</p> <ol style="list-style-type: none"> 1. Create a connectivity agent group. See Create an Agent Group. 2. Download and run the on-premises connectivity agent installer in your local environment. During installation setup, associate the on-premises connectivity agent with the agent group. See Download and Run the Connectivity Agent Installer or Use the Connectivity Agent in High Availability Environments. <p>After you install the connectivity agent, you can create a connection that uses the agent group and its associated connectivity agent. See Create a Connection with an Agent Group.</p>
4	Configure a private endpoint	<p>Required if: You need to connect to private resources and must route the traffic through a private channel.</p> <p>Not needed if: You don't need to connect to private resources, or you need to connect to private resources but don't need to route traffic through a private channel.</p> <p>Learn more: Connect to Private Resources in <i>Provisioning and Administering Oracle Integration 3</i>.</p> <p>Get started:</p> <ol style="list-style-type: none"> 1. Complete the prerequisites for configuring a private endpoint. See Prerequisites for Configuring a Private Endpoint in <i>Provisioning and Administering Oracle Integration 3</i>. 2. Configure the private endpoint. See Configure a Private Endpoint for an Instance in <i>Provisioning and Administering Oracle Integration 3</i>. <p>Later, when you create connections, you choose whether they use the private endpoint to connect to applications.</p>

 **Note:**

A connection that doesn't use the connectivity agent or a private endpoint is routed through the public internet.

After completing the prerequisites, you can start designing integrations. See [Workflow for Developing an Integration](#).

Workflow for Developing an Integration

To develop an integration, first create a project to work in. Then, create the integration and all its components, and specify the steps that the integration completes.

Before completing these tasks, make sure you complete all prerequisites. See [Prerequisites for Developing Integrations](#).

Step	Task	More information
1	Create a project	<p>Create one or many integrations in each project. Most organizations group related integrations in a single project.</p> <p>Learn more: Get Started with Projects.</p> <p>Get started: Create or Import a Project.</p>
2	Create an integration	<p>Oracle recommends designing integrations in projects, but you can also design an integration outside of a project.</p> <p>When you create the integration, add trigger and invoke connections, assign business identifiers, and then map the data between the connections.</p> <p>Learn more: About Integrations.</p> <p>Get started: Create, Import, or Copy an Integration into a Project or Create an Integration.</p>
3	Choose the applications to connect	<p>Connect to applications using connections. Create one connection for each application that you need to integrate. Typically, an administrator creates connections.</p> <p>You can use a connection in one or many integrations.</p> <p>Learn more: About Connections.</p> <p>Get started: Add and Share a Connection Across a Project.</p>
4	Specify which data to move and how to move it	<p>Specify the data to move and the way to move the data by adding trigger and invoke connections to an integration. At that time, you specify your requirements, including how to process the data.</p> <p>Learn more: About Connections.</p> <p>Get started: Understand Trigger and Invoke Connections.</p>
5	Map different terms that describe the same item (required for only some integrations)	<p>Use lookups to associate different values for the same item. For example, two applications might have different gender codes or country codes. Lookups ensure that the correct values appear in the target application.</p> <p>Required if: The applications you're connecting share and represent information differently.</p> <p>Learn more: About Lookups.</p> <p>Get started: Add a Lookup to a Project.</p>
6	Upload JavaScript for use in your integration (optional)	<p>When you need to call JavaScript from an integration, you can import the JavaScript file.</p> <p>Learn more: Use Libraries to Manage Integration Functions.</p> <p>Get started: Import a JavaScript Library in a Project.</p>

Step	Task	More information
7	Design the business logic of your integration	<p>Every integration performs one or more activities, such as looping over elements and adding routing expressions. Add actions to perform the activities that the integration must complete.</p> <p>Get started:</p> <ul style="list-style-type: none"> To review the wide range of available actions, see Add Actions to an Application Integration. In addition to the actions that define the business logic of the integration, consider including the following actions as best practices: <ul style="list-style-type: none"> Add email notifications to the integration You can use email notifications to send confirmations or inform you about faults that have occurred. Oracle Integration sends the email notifications at specific processing points for an integration, and the notifications help you quickly identify issues with important business processes. See Send Notification Emails During Stages of the Integration with a Notification Action. Add global fault handling to the integration A fault handler translates cryptic error messages into actionable tasks so your business can keep running smoothly. If the APIs you're connecting to return error messages that are difficult to understand, consider adding global fault handling. See Add Global Fault Handling to Integrations. Add fault handlers to a collection of actions and invokes When you move some business logic, including actions and invokes, into a scope, you can add fault handlers to the business logic. See Manage a Group of Actions and Fault Handlers with a Scope Action.
8	Select the payload fields to track in runtime	<p>You track fields in messages during runtime by assigning business identifiers to the fields.</p> <p>Get started: Assign Business Identifiers for Tracking Fields in Messages.</p>
9	Specify when your integration runs (schedule integrations only)	<p>Schedule integrations run by the clock. Specify when to run the integration and more details when you schedule the integration.</p> <p>Get started: Define the Integration Schedule.</p>

Next, deploy the integration to a higher environment, if needed. See [Workflow for Deploying Automation Solutions](#). Or, you can test the integration in your development environment. See [Workflow for Testing Integrations](#).

Workflow for Testing Integrations

You can test integrations in your development environment or in a higher environment, such as testing or production. When testing, check the integrations for errors and ensure that they run as expected. You should understand the goals and business logic of an integration so that you can identify issues quickly.

If you need to deploy to a higher environment before testing, see [Workflow for Deploying Automation Solutions](#).

Step	Task	More information
1	Activate the project or the integrations in the project	<p>You have several options:</p> <ul style="list-style-type: none"> In any environment, you can activate the project, which activates all integrations in it. The Debug tracing level isn't available when you activate a project, but you can update the tracing level for individual integrations after activation. See Update the Tracing Level of Integrations in a Project. If you're working in your development environment, you can also activate integrations individually. When you activate integrations individually, you can choose the tracing level for each integration. See Activate an Integration. <p>Note: When you activate an integration, you choose the level of tracing information to log to the activity stream. Consider the following factors when choosing a tracing level:</p> <ul style="list-style-type: none"> To see a detailed log of every activity that the integration completes, such as when you're testing, select Debug (Not recommended) for the tracing level. After 24 hours, Debug (Not recommended) is automatically reset to Production. Debug tracing provides the most robust level of information in the activity stream and can help you find mistakes. However, Oracle recommends not using debug tracing in production for several reasons. Debug tracing captures a lot of data and can cause performance issues for an integration, and it might expose sensitive data. If you're testing the invokes or the trigger of the integration, you don't need to use debug tracing. Instead, the Audit tracing level captures payloads for these activities.
2	Run the integration	<p>To run an integration, invoke the endpoint URL for the integration. You have several options for invoking an endpoint URL:</p> <ul style="list-style-type: none"> For any integration You can trigger any integration from outside Oracle Integration. Copy the endpoint URL from within Oracle Integration, and use it in a client application to run the integration. For instance: <ul style="list-style-type: none"> If the trigger for an integration is the creation of a specific event in Salesforce, create the event in Salesforce. If an integration trigger uses the SOAP Adapter, use a SOAP user interface to trigger the endpoint URL. If an integration trigger is based on the REST Adapter, use Postman to trigger it, or see the next bullet. For schedule integrations and integrations that use trigger-based REST Adapter connections Run these integrations from within Oracle Integration. See Run a Schedule Integration Immediately and Test REST Adapter Trigger Connection-Based Integrations in Projects.

Step	Task	More information
3	Check the activity stream for errors	<p>Verify that the instance of the integration is finished. See Monitor Integrations in a Project.</p> <p>Next, verify that the integration finished running, and check the activity stream for errors. The activity stream depicts the message flow in the integration instance and indicates where errors occurred, if any. The activity stream contains all the information that you need to troubleshoot problems with your integrations.</p> <p>To learn more, see Learn About the Payloads in the Activity Stream.</p> <p>You can open the activity stream from several pages:</p> <ul style="list-style-type: none"> If you ran the integration from within Oracle Integration, open the activity stream from the Configure and run page, which appears after you run the integration. See Test REST Adapter Trigger Connection-Based Integrations in Projects. If the integration is in a project, open the activity stream from the Instances tab, which is on the Observe tab. See Track Instances in a Project. If the integration is outside a project, or if it's in a project that you have access to, open the activity stream from the Instances page, which is within Observability. See View Error Messages and the Message Payload in the Activity Stream of Failed Instances.
4	If an error occurred when the integration ran, address it and take the appropriate next steps	<p>When the activity stream contains one or more errors, the integration isn't running as expected.</p> <ol style="list-style-type: none"> Identify the cause of the issue by reviewing the activity stream. Get more information about an error on the Instance Details page. This page provides a graphical view of the integration flow and also includes the activity stream. <ul style="list-style-type: none"> If you're working in a project, see Track Instances in a Project. If you're not working in a project, see View Error Messages and the Message Payload in the Activity Stream of Failed Instances. Address the error as needed. For example: <ul style="list-style-type: none"> Update the connection or the configuration of the lookup. Fix the integration in your development environment and redeploy it. Run the integration again, and repeat these steps as needed until the integration runs without errors.
5	Verify that the integration works as expected	<p>For example:</p> <ul style="list-style-type: none"> Verify that the integration creates records as expected in the target application. Verify that the integration handles expected failures, errors, and invalid data as expected.

Workflow for Deploying Automation Solutions

You have several opportunities to deploy to higher environments. Many organizations deploy to a testing environment to simulate real-world usage, including testing under load. Additionally, everyone should deploy to a production environment when you're ready to go live. The steps you follow are the same for both environments.

 **Note:**

Complete these steps only if you're promoting to a higher environment and are working in a project:

- If you use a single environment for development and testing, these steps apply only when you're promoting to your production environment. For guidance on testing, see [Workflow for Testing Integrations](#).
- If you're not using projects, follow your organization's standard procedures for deploying integrations to higher environments.

If you haven't developed the integration yet, see [Workflow for Developing an Integration](#).

Step	Task	More information
1	Create a project deployment	<p>Work in your development environment. Include the integrations and their versions in the project deployment.</p> <ul style="list-style-type: none">• To work in the user interface, see Create and Manage a Project Deployment.• To use the REST API, use the <code>deployments</code> endpoint. See Deployments REST Endpoints in <i>REST API for Oracle Integration 3</i>.
2	Export the project deployment	<p>Export the project deployment to the location of your choice, such as your hard drive or a location on your network.</p> <ul style="list-style-type: none">• To work in the user interface, you have the following options:<ul style="list-style-type: none">– Export the entire project, including all of the integrations and components. See Export a Project.– Export a specific project deployment, including only the integrations and components that are part of the deployment. See Create and Manage a Project Deployment.• To use the REST API, use the <code>archive</code> endpoint. See Export a Project in <i>REST API for Oracle Integration 3</i>.

Step	Task	More information
3	Import the project deployment into the higher environment	<p>Sign in to the environment to which you're deploying the integrations, such as the testing or production environment. Then, import the exported project into the environment.</p> <ul style="list-style-type: none"> • To work in the user interface, see Create or Import a Project. • To use the REST API, use the <code>archive</code> endpoint. See Import(Add) a Project in <i>REST API for Oracle Integration 3</i>. <p>Note: To synchronize the project with a source control system, use the <code>archive</code> endpoint to export the entire project. See Export a Project in <i>REST API for Oracle Integration 3</i>.</p>
4	Configure the properties for the connections in the integrations	<p>You must update your connection properties so that they reflect the higher environment that you imported into.</p> <p>See Configure Connection Properties and Security Properties.</p>

Next, it's time to test the integration. See [Workflow for Testing Integrations](#).

Workflow for Monitoring Integrations

Monitoring tasks typically fall into one of three categories: initial, daily, and occasional.

- [Initial Monitoring Tasks](#)
- [Daily Monitoring Tasks](#)
- [Occasional Monitoring Tasks](#)

To learn more about observability in Oracle Integration, see [Get Started with Observability](#).

Initial Monitoring Tasks

Before an integration goes live in production, optimize your monitoring workflow by completing a few tasks.

You might need to complete some tasks again when your organization prepares to release a major update to an integration.

Task	More information
Complete performance testing	See Measure an Integration's Performance .
Complete load testing	<p>Oracle recommends checking the scalability of a new integration by performing load testing. Checking the performance of an integration under different loads helps you understand how the integration will perform in a real-world scenario. For instance, how will your order creation integration handle Black Friday loads?</p> <p>After testing the integration under load, measure its performance. See Measure an Integration's Performance.</p>

Task	More information
Follow your organization's standard operating procedures for preparing for an integration to go live	<p>For example, document the following information in a location that stakeholders can access:</p> <ul style="list-style-type: none"> • Name of the integration • Overview of the business process that the integration automates • Applications that are involved • Contact names, including backups, for who can help with failures and who should be informed in the event of a failure • Name of the connectivity agent that the integration uses, if any <p>See View the Dashboard.</p>
If your organization maintains custom monitoring tools, determine whether the new integration requires updates to the tools	See About Building Your Own Observability Tools .
For schedule integrations, make sure the start time doesn't conflict with other integrations	<p>For example, don't run twenty schedule integrations at the close of business. Instead, stagger the integrations by at least a few minutes.</p> <p>See View the Calendar of Schedule Integration Runs.</p>
Set up notifications	<p>Notifications provide daily or hourly emails about failures or the overall health of all your integrations.</p> <p>See Send System Status Reports with Notification Emails.</p>
Determine when responsibility for the integration changes hands	<p>If you're responsible for monitoring an integration that you built, this task doesn't apply to you.</p> <p>However, if you're responsible for monitoring someone else's integration, work with the integration developer to determine a hand-off schedule. For instance, do you take ownership when the integration goes live, or only after the integration has run successfully in production for a period of time?</p> <p>Setting expectations from the beginning helps ensure the successful launch of a new business process automation.</p>
(When your organization creates a new version of a live integration)	After you version and update an integration, verify that the new version of the integration performs as expected. For example, you can compare the average running times of the old and new integrations.
Verify that the new version doesn't contain any performance regressions	See Determine an Integration's Typical Processing Times .

Daily Monitoring Tasks

Your organization's requirements how often you perform these monitoring tasks. Typically, organizations perform these tasks at least daily, often at predetermined times each day.

Task	More information
Determine whether any integrations experienced errors	See Find Integrations That Experienced Errors .
Determine whether any integrations timed out or were aborted (canceled)	See Find Integrations That Timed Out or Were Aborted .

Task	More information
Determine whether an integration has experienced performance issues	Troubleshoot slow-running integrations and verify that integrations meet your organization's performance requirements. See Measure an Integration's Performance .
Identify the most recent updates to an integration	Consider a scenario in which an integration that has previously run as expected is suddenly not working as expected. An inadvertent update could be to blame. Check the integration's audit history to understand who updated it, what changes they made, and when they updated it. For instance, if unscheduled changes occurred, discuss the update with the developer and determine whether the changes need to be rolled back. See Check the Audit History for an Integration or Other Component .
Determine why fewer integration instances are running than expected	See Fewer Integration Instances Ran Than Usual .
Determine whether all connectivity agents are available	If a connectivity agent isn't available, any integrations that use the agent fail. Inform the administrator who is responsible for the connectivity agent if any agents go down. See Monitor Agents .
Identify the connections that cause the most errors	See Find the Connections with the Most Errors .

Occasional Monitoring Tasks

Routine analysis work supports operational excellence. Use your organization's operating procedures to determine how often you complete these tasks. For example, you might complete some tasks weekly, monthly, or quarterly.

Task	More information
Verify that integrations continue meeting your organization's performance requirements	See Measure an Integration's Performance .
Ensure that the tracing level is set correctly for integrations	See Check an Integration's Tracing Level .
Make sure that your organization's active integrations stay below the limit	On the Dashboards page, see the numbers in the Active integrations box. To learn more about the service limits, <i>Service Limits in Provisioning and Administering Oracle Integration 3</i>
Confirm that your schedule integrations start at staggered times	If too many schedule integrations start at the same time, they might experience contention. Ease the contention by staggering the start times. See View the Calendar of Schedule Integration Runs .
See a snapshot of the current status of the components your organization has developed	See See the Current Status of All Integration Components .
Update the information you've documented for integrations	For example, if you document the contacts and stakeholders for an integration, periodically review the information and make updates for organizational changes and employee turnover.

Tours and Tutorials

The following tours and tutorials introduce you to integrations and show the ease of designing, activating, invoking, and monitoring an integration.

Tours

Tours walk you through the design-time and runtime aspects of integrations and require no hands-on activities.

 **Note:**

Sample recipes and accelerators are available for download from the Integration Store. See Get Familiar with the Home Page in *Getting Started with Oracle Integration 3* and [Recipes and Accelerators](#) on the Oracle Help Center.

Tour	Complexity	Description
Explore the Navigation Pane	Low	Learn how the navigation menu helps you find your way around.
Tour: Explore an Integration That Sends a Hello World Message	None	Oracle Integration includes a recipe for a Hello World integration, which uses a simple log notification action with a REST Adapter in an application integration. In this tour, you review the design-time and runtime aspects of a Hello World integration.
Tour: Explore a Multiple Verbs and Resources Invoke Integration	Medium	Oracle Integration includes a recipe that creates and invokes a REST web service with multiple HTTP verbs and resources. In this tour, you review the design-time and runtime aspects of this integration.

Tutorials

In a tutorial, you build everything you need for an integration from start to finish.

Tutorial	Complexity	Description
Build Your First Integration from Scratch	Medium	Design, run, and monitor a Hello World integration from start to finish.
Build Your Second Integration from Scratch	Medium	Design, run, and monitor an integration that receives a weather report for a city. This integration invokes a weather application with the REST Adapter.
Tutorial: Read Files and Write the Files to File Server	Medium	Learn how to use the File server action to read one or more files that are on an FTP server and then write the files to File Server in Oracle Integration.

LiveLab Workshops

In a hands-on, self-directed workshop, gain expertise in a couple hours.

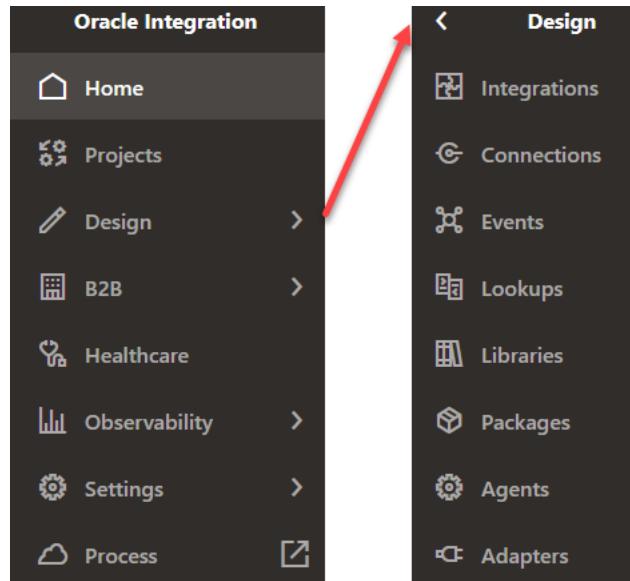
- **LiveLabs for new users**
 - Get Started with Oracle Integration 3
 - Get Started with Oracle Integration B2B
 - Get Started with Healthcare
 - Get Started with Process Automation
 - Get Started with Rapid Adapter Builder
 - Get Started with Robotic Process Automation (RPA)
- **LiveLabs for Oracle Fusion Cloud Applications**
 - ERP Cloud BIP Report
 - ERP Cloud Bulk Extract
 - ERP Cloud Business Events
 - ERP Cloud FBDI Import
 - ERP Cloud Real-Time Synchronization
 - HCM Cloud Bulk Data Import
 - HCM Cloud Directory Synchronization
- **Additional LiveLabs**
 - Automate Invoice Processing with OCI Document Understanding
 - Decouple Producers and Subscribers Using Events
 - Understand the REST Adapter
 - Use Built-In Actions to Harness OCI Services

Explore the Navigation Pane

Use the navigation pane to access all the pages, tools, dashboards, and settings in Oracle Integration.

When you start Oracle Integration, the navigation pane is hidden. To view the navigation pane, click **Show/Hide navigation menu**  in the top corner of the Home page.

The menu displayed in the navigation pane depends on your assigned role, your selection, and your location in Oracle Integration. The navigation pane may display the main menu or a submenu.



Browse the menu and submenu selections, which give you access to the Oracle Integration runtime environment, design-time environment, development tools, and administration settings.

Menu Selections	More Information
Home	Access the launch pad and high-level dashboard for Oracle Integration, search for recipes and accelerators, and open recently worked on projects.
Projects	<p>Launch the page to create integration projects. An integration project consists of self-contained assets (integrations, connections, lookups, and JavaScript libraries) bundled into a solution that can be developed, managed, and monitored as a single unit from a single workspace.</p> <p>For more information about working with projects, see the following topics:</p> <ul style="list-style-type: none">• About Integration Projects• Designing, Managing, and Monitoring Integrations in Projects

Menu Selections	More Information
 Design > <ul style="list-style-type: none">  Integrations  Connections  Events  Lookups  Libraries  Packages  Agents  Adapters 	<p>Launch the design-time environment for integrations. Configure connections, and create and activate integrations. Work with lookups, libraries, packages, agents, and adapters.</p>
 B2B > <ul style="list-style-type: none">  Documents  Schemas  Trading partners  Host profile 	<p>Exchange business documents between Oracle Integration and a trading partner securely and reliably.</p>
 Healthcare > <ul style="list-style-type: none">  HL7 messages  Schemas 	<p>Exchange messages between Oracle Integration and healthcare organizations that use HL7 and FHIR.</p>

Menu Selections	More Information
 Observability > <ul style="list-style-type: none"> <li data-bbox="393 403 649 430"> Dashboards <li data-bbox="393 481 649 508"> Integrations <li data-bbox="393 559 649 587"> Subscriptions <li data-bbox="393 616 649 644"> Agents <li data-bbox="393 694 649 722"> Instances <li data-bbox="393 772 649 800"> Errors <li data-bbox="393 830 649 857"> Future runs <li data-bbox="393 908 649 935"> B2B tracking <li data-bbox="393 965 649 992"> Fusion Applications 	<p>Monitor integrations in the runtime environment.</p> <p>For more information about the submenu selections, see the following topics:</p> <ul style="list-style-type: none"> • Dashboards: View the Dashboard • Integrations: Monitor the Message Processing Status of Integrations • Subscriptions: Monitor Event Subscription Integration Status • Agents: Monitor Agents • Instances: Track Integration Instances • Errors: Manage Errors • Future runs: View the Scheduled Integration Runs Calendar • B2B tracking: Track B2B Messages • Fusion Applications: Diagnose and Manage Event-Based Oracle Fusion Applications Integrations
 Settings > <ul style="list-style-type: none"> <li data-bbox="393 1184 605 1212"> Certificates <li data-bbox="393 1241 605 1269"> Notifications <li data-bbox="393 1298 605 1326"> Tracing <li data-bbox="393 1355 605 1383"> Schedule <li data-bbox="393 1412 605 1440"> Security <li data-bbox="393 1469 605 1497"> Storage <li data-bbox="393 1526 605 1554"> Import and Export <li data-bbox="393 1584 605 1611"> File Server 	<p>Configure settings for security certificates, notifications, tracing, schedules, time limits for inactive sessions, storage buckets, and import and export of design-time metadata. Access File Server settings for preferences, users, groups, and folders.</p>
	<p>For more information about the submenu selections, see the following topics in <i>Using Integrations in Oracle Integration 3</i>:</p> <ul style="list-style-type: none"> • Certificates: Manage Security Certificates • Notifications: Configure Notification Emails • Tracing: Change the Tracing Level on an Active Integration • Schedule: Globally Change the Submitter of Integration Schedules • Security: Set the Time Limit for Inactive Sessions • Storage: Step 2: Configure the Instance Object Storage Bucket • Import and Export: Step 3: Export and Import Design-Time Metadata Between Instances <p>For File Server settings, see Administer File Server.</p>
 Visual Builder > <ul style="list-style-type: none"> <li data-bbox="393 1727 605 1755"> Cloud 	<p>Launch Visual Builder to create and publish web and mobile applications.</p> <p>For more information about working with Visual Builder, see Get Started with Visual Builder in <i>Developing Applications with Oracle Visual Builder in Oracle Integration 3</i>.</p>
 Process > <ul style="list-style-type: none"> <li data-bbox="393 1890 605 1917"> Cloud 	<p>Launch Oracle Cloud Infrastructure Process Automation to rapidly design, automate, and manage business processes in the cloud.</p> <p>For more information about working with Oracle Cloud Infrastructure Process Automation, see Use Process Automation with Oracle Integration 3 in <i>Using Oracle Cloud Infrastructure Process Automation</i>.</p>

Tour: Explore an Integration That Sends a Hello World Message

This tour provides a walkthrough of a Hello World integration, including steps to monitor the integration.

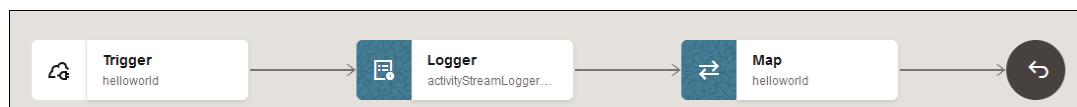
To see a similar Hello World integration in your instance, you can install a Hello World recipe. Or, simply review the screenshots and steps below to understand the integration and your workflow. You can also build a Hello World integration from scratch. See [Tutorial: Build Your First Integration From Scratch](#).

Step 1. View the Integration Design

This integration is based on an application integration. It triggers the REST Adapter when you specify the integration name. The integration creates a logging message and sends it to the activity stream for logging.

1. In the navigation pane, click **Design**, then **Integrations**.

This integration consists of the following:



- A REST Adapter is configured as a trigger (inbound) connection in the integration. The REST Adapter is configured with a resource endpoint of `/input/{HelloWorld}` and a GET operation. The REST Adapter is configured to receive a message and respond with a message. The REST Adapter is triggered when you specify an integration name on the Configure and run page.
- A logger action is configured with a `concat` function that accepts the integration name you specify on the Configure and run page and concatenates it to the message.

`concat('Integration name is :', name)`

- A mapper maps the source value to the target message element.
- A response is returned.

Step 2. View How to Run the Integration

1. Hover your cursor over **Hello World**, then click **Activate**
2. Click **Activate** when prompted.
3. Wait for the icon to turn green and the word **Active** to appear in the **Status** column, indicating that the integration is activated.
4. Hover your cursor over **Hello World**, then click and select **Run**. The Configure and run page is displayed. This page is available only for REST Adapter-trigger based integrations and scheduled integrations.
5. Enter **Hello World**, and click **Run**.

Configure and run

Configure request properties, then run the integration and verify the response. [Click here to read more.](#)

Endpoint Metadata Track Instances

GET
/ic/api/integration/v1/flows/rest/HELLO_WORLD/1.0/input/Hello World

Request

URI Parameters Headers Body cURL Integration Properties

Helloworld Hello World

Step 3. View the Movement of Messages Through the Integration

The Activity Stream panel slides open on the Configure and run page. The activity stream shows the status of message processing through the major milestones of the integration.

1. Expand the logger action to see the message.

Activity Stream

Instance Id: OY3StQrZEe2_OdIYmijew [Copy](#)

Tracing level: **Production**

Sat Jul 23 04:46:00.813 PM MDT 2022, 47ms
Wire Message received by Trigger helloworld

04:46:00.860 PM, 1ms
Message started processing

04:46:00.861 PM, 0ms
Logger activityStreamLogger_1634894950

Integration name is :Hello world

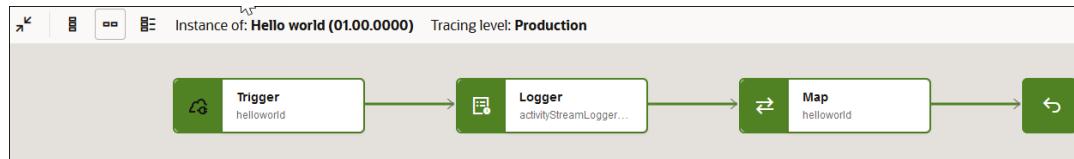
04:46:00.861 PM, 2ms
Data Mapping completed

04:46:00.863 PM, 4ms
Message sent to reply

04:46:00.867 PM
Reply to helloworld completed

Step 4. View How to Monitor the Integration

1. Click the instance ID at the top of the Activity Stream panel. The Instance Details page provides a graphical view of the integration flow. The same Activity Stream panel is also visible on the right.



2. Click < to return to the Configure and run page.
3. Click **Track Instances** and note that the **Hello World** integration instance is listed as **Succeeded** in the **Status** column.

1 Instances				
Primary Identifier	Instance Id	Status	Business Identifiers	Duration
HelloWorld: Hello World Hello world 1.0.0	OY3StQrZEE...	Succeeded		Received 42 minutes ago Succeeded 42 minutes ago Duration 54 milliseconds

4. In the navigation pane, click **Observability**, then **Integrations**.

The message is listed as successfully received and processed without any errors.

1026 Integrations						
Name	Received	Processed	Succeeded	Errored	Aborted	Last Scheduled By
>Hello world (1.0) Last Message: Just now	1	1	1	0	0	

Learn More About The Features in this Tour

- [Create Application Integrations](#)
- REST Adapter Capabilities
- [Log Messages with a Logger Action](#)
- [Test Integrations from Outside the Integration Canvas](#)
- Getting Started with the Mapper

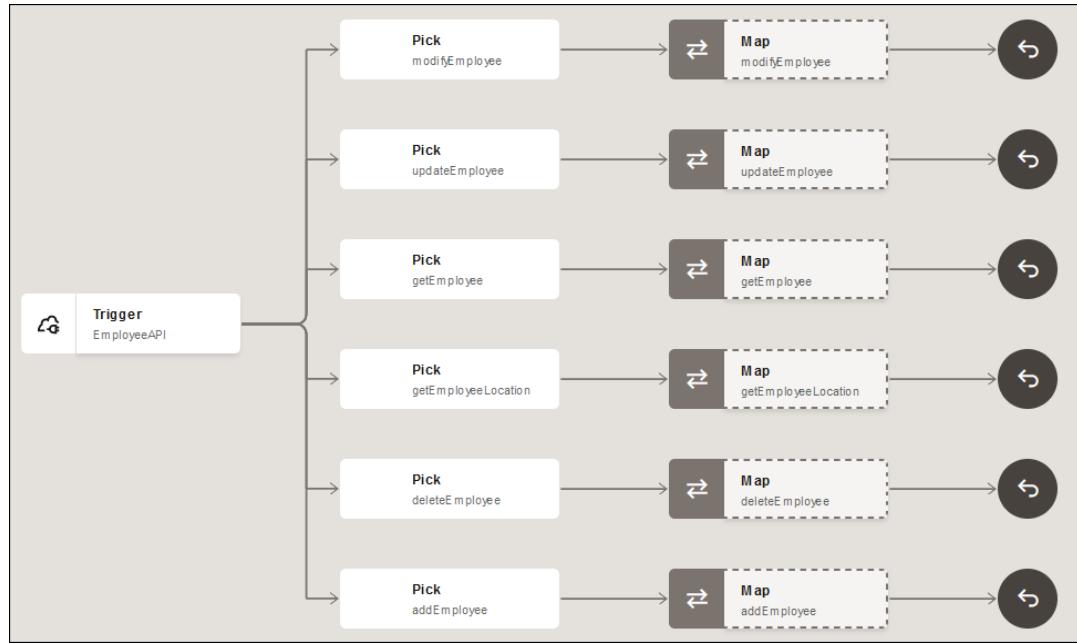
Tour: Explore a Multiple Verbs and Resources Invoke Integration

This example shows the Use Multiple Verbs and Resources Invoke recipe, which demonstrates how to use multiple HTTP verbs and resources configured in the REST Adapter trigger connection of an integration. This recipe is available from the Integration Store.

You can configure a single REST Adapter to include multiple HTTP verbs such as GET, PUT, POST, PATCH, and DELETE. You can also configure the REST Adapter with multiple resources. For this example, a single /emp resource and an /emp/loc subresource are used.

Step 1. View the Integration Design

1. In the navigation pane, click **Design**, then **Integrations**.
This integration consists of the following:



- A REST Adapter connection (**EmployeeAPI**) is configured as a trigger (inbound) connection in the integration. The REST Adapter connection is configured with multiple operations: **modifyEmployee** (PATCH), **updateEmployee** (PUT), **getEmployee** (GET), **getEmployeeLocation** (GET), **deleteEmployee** (DELETE), and **addEmployee** (POST).

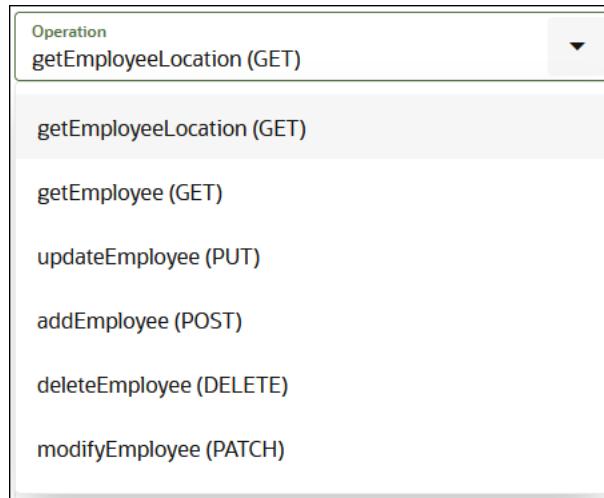
Operation	Resource	HTTP Method	Edit/Delete
modifyEmployee	/emp	PATCH	<input type="checkbox"/>
updateEmployee	/emp	PUT	<input type="checkbox"/>
getEmployee	/emp	GET	<input type="checkbox"/>
getEmployeeLocation	/emp/loc	GET	<input type="checkbox"/>
deleteEmployee	/emp	DELETE	<input type="checkbox"/>
addEmployee	/emp	POST	<input type="checkbox"/>

- The REST Adapter is triggered when you specify the URL.
- Each operation is shown as a different path that branches out from the trigger connection.
- Each branch has a map and a reply action configured.

- The tracking configuration for each branch can be viewed by clicking **Actions ...**, then **tracking** on each pick action.

Step 2. View How to Run the Integration

- In the navigation pane, click **Design**, then **Integrations**.
- Hover your cursor over **Sample Multiple Verbs and Resources Invoke**, then click **Activate** .
- Click **Activate** when prompted.
- Wait for the icon to turn green and the word **Active** to appear in the **Status** column, indicating that the integration is activated.
- Hover your cursor over **Sample Multiple Verbs and Resources Invoke**, then click **Actions ...** and select **Run**. The Configure and run page is displayed with details about running, tracking, and testing the integration.
- Click the **Operation** drop-down list to view all configured options and HTTP verbs.
- From the drop-down list, select the operation to perform. For this example, **addEmployee (POST)** is selected.



The section following the drop-down list shows all the options available for the selected operation such as **URI Parameters** and **Body**.

- Click **Run** to run the selected operation.

Step 3. View the Movement of Messages Through the Integration

The Activity Stream panel slides open on the Configure and run page. The activity stream shows the status of message processing through the major milestones of the integration.

Activity Stream

Instance Id: aOr7Cwt2Ee2_OdIYmijJew [Copy](#)

Tracing level: Production

Sun Jul 24 11:31:11.263 AM MDT 2022, 267ms
Message processed by Trigger Pick Receive for Operation EmployeeAPI_addEmployee

11:31:11.530 AM, 1ms
Message started processing

11:31:11.531 AM, 0ms
Data Mapping completed

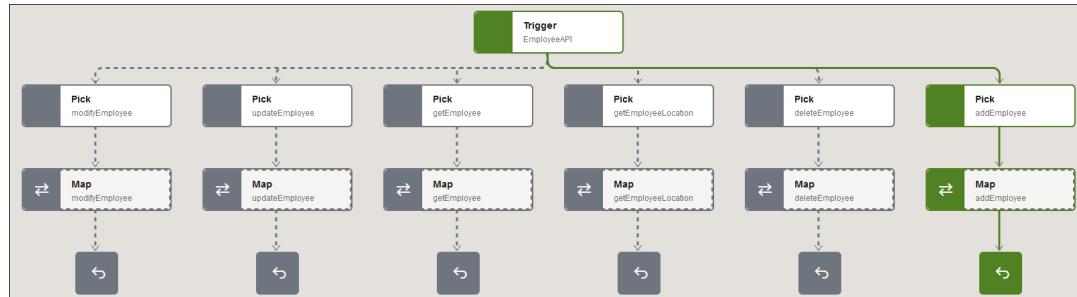
11:31:11.531 AM, 6ms
Message sent to reply

11:31:11.537 AM, -5ms
Reply to EmployeeAPI_addEmployee completed

11:31:11.532 AM
Processing completed successfully

Step 4. View How to Monitor the Integration

- Click the instance ID at the top of the Activity Stream panel. The Instance Details page provides a graphical view of the integration flow. The path taken by this instance is highlighted in green, indicating it was successful. The same Activity Stream panel is also visible on the right.



- Click < to return to the Configure and run page.
- Click **Track Instances** and note that the **Sample Multiple Verbs and Resources Invoke** integration instance is listed as **Succeeded** in the **Status** column.

4 Instances					
Primary Identifier	Instance Id	Status	Business Identifiers	Duration	
Primary: undefined Oracle Sample Multi... 1.0.0	aOr7Cwt2Ee2_OdIYmijJew...	Succeeded		Submitted Succeeded Duration	5 minutes ago

- In the navigation pane, click **Observability**, then **Integrations**.
The message is listed as successfully received and processed without any errors.

Name	Received	Processed	Succeeded	Errored	Aborted	Last Scheduled By
Oracle Sample Multiple Verbs and Resources Invoke (1.0) Last Message: 2 minutes ago	1	1	1	0	0	

Learn More About the Features in This Tour

- [Create Application Integrations](#)
- [Receive Requests for Multiple Resources in a Single REST Adapter Trigger with a Pick Action](#)
- [Test Integrations from Outside the Integration Canvas](#)
- [Getting Started with the Mapper](#)

Tutorial: Build Your First Integration From Scratch

Let's walk through how to create, design, activate, run, and monitor your first integration completely from scratch. You work inside a project. A project contains related resources, including integrations, connections, and more. You can design, manage, and monitor everything in one place.

- [What Do I Learn from this Tutorial?](#)
- [Step 1. Create a Project in Which to Design, Manage, and Monitor Your Integration](#)
- [Step 2. Create an Integration](#)
- [Step 3. Create a Trigger Connection to Allow Your Integration to Receive Messages](#)
- [Step 4. Specify How the Trigger Connection Receives Messages in the Integration](#)
- [Step 5. Design How the Integration Receives Messages](#)
- [Step 6. Define a Business Identifier to Track Message Delivery at Runtime](#)
- [Step 7. Activate and Run the Integration](#)
- [Step 8. Monitor Message Delivery Status at Runtime](#)
- [Learn More About The Features in this Tutorial](#)

What Do I Learn from this Tutorial?

This tutorial walks you through performing the following tasks:

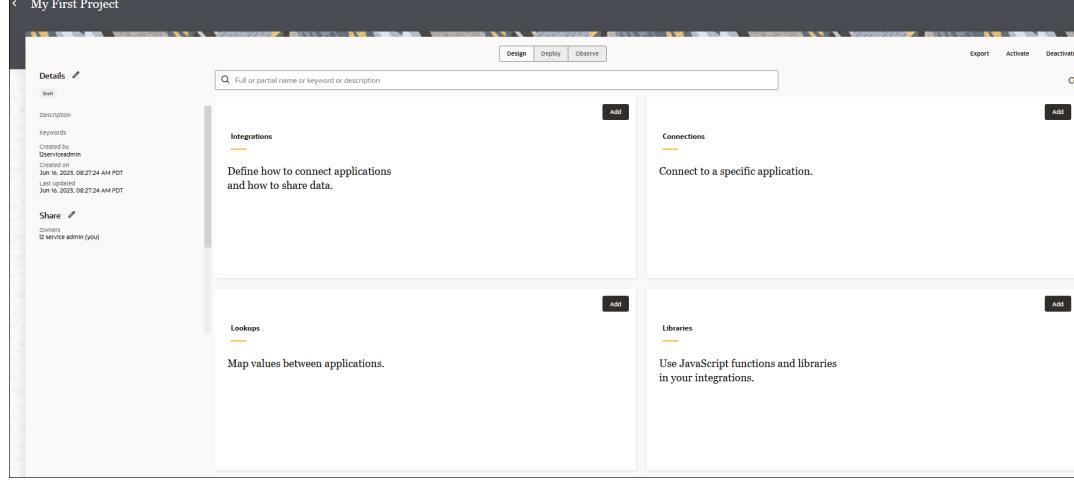
- Creating a project in which you perform all design, management, and monitoring tasks.
- Creating and configuring an adapter connection that lets your integration receive inbound messages.
- Creating and configuring an integration that takes either of two paths to process the inbound message:
 - If an email address is specified
 - If an email address is not specified
- Running (triggering) the integration by specifying a simple message consisting of a name, an optional flow ID, and an email address.
- Monitoring the results of the integration run.

Step 1. Create a Project in Which to Design, Manage, and Monitor Your Integration

You manage the entire life cycle of an integration from within a project.

1. In the navigation pane, click **Projects**.
2. Click **Add**, then select **Create**.
3. In the **Name** field, enter **My First Project**.
4. Leave all other fields as they are, then click **Create**.

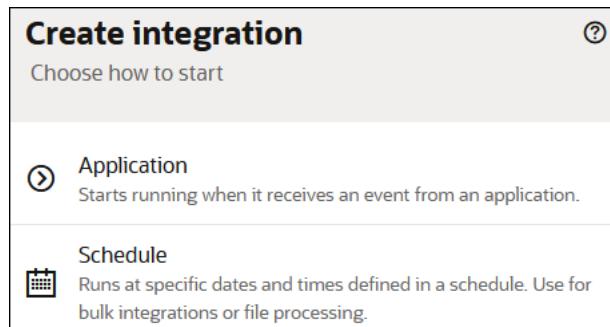
The page for your first project is displayed. Your project contains sections for creating, designing, deploying, and monitoring your integration.



Step 2. Create an Integration

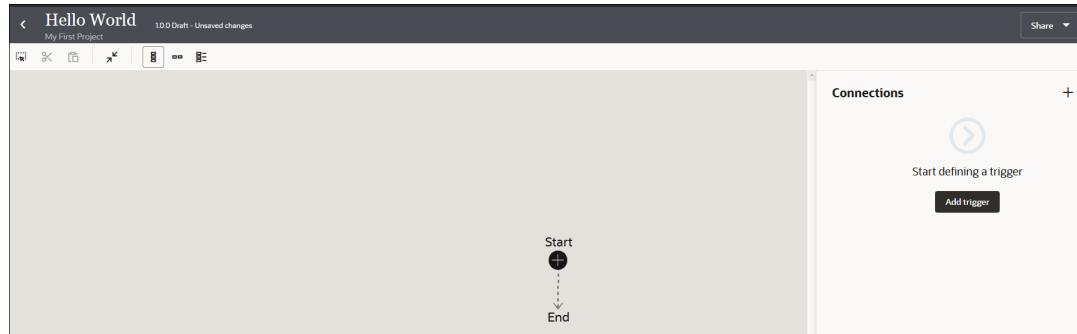
Let's create your first integration inside the project.

1. In the **Integrations** section, click **Add**.
The Add integration panel opens.
2. Click **Create**, then select **Application**.



3. In the **Name** field, enter **Hello World**.
4. Leave all other fields as they are, then click **Create**.

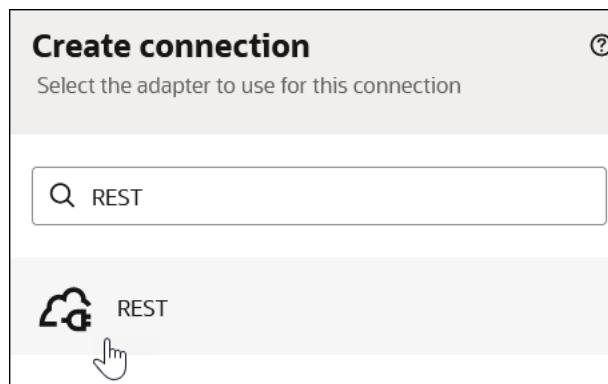
The integration canvas opens with an option for creating a trigger connection. The trigger connection lets your integration receive inbound messages from an application.



Step 3. Create a Trigger Connection to Allow Your Integration to Receive Messages

Now let's create a trigger connection inside the integration canvas. The trigger (source) connection allows your integration to receive inbound messages from an application.

1. In the Connections panel, click **Add trigger**.
The Create connection panel opens.
2. Enter REST in the search field.
3. Select **REST** as the adapter connection to use. For this tutorial, the REST Adapter processes inbound messages sent to Oracle Integration.



The page for configuring the REST Adapter opens.

- a. Enter the following information.

Field	Description
Name	Enter Trigger1.
Role	Select Trigger . Roles perform the following tasks. <ul style="list-style-type: none">• Trigger: Enables an inbound application (endpoint) to trigger (run) the integration.• Invoke: Enables your integration to invoke an outbound application (endpoint).

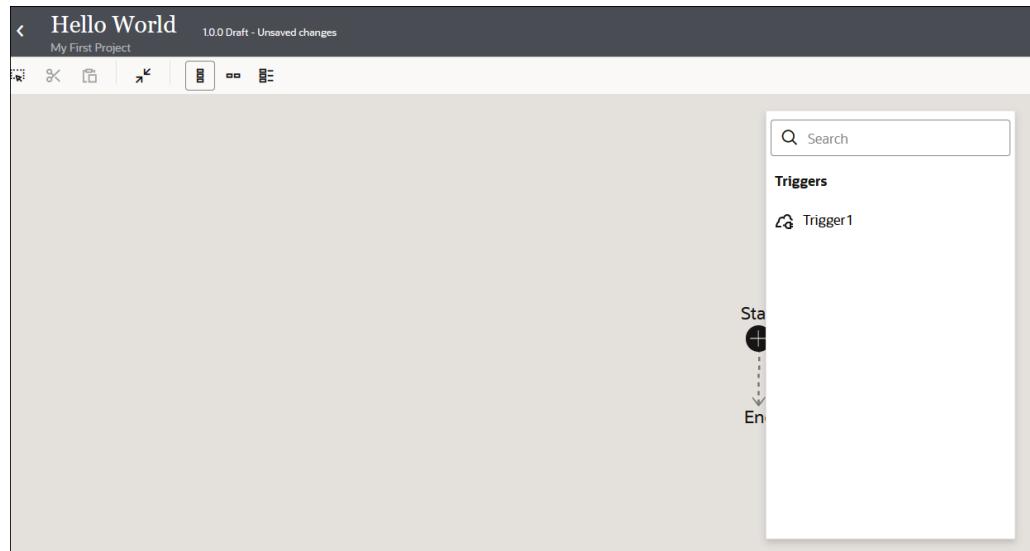
- b. Ignore all other fields, and click **Create**.

The page for configuring security and delivery methods for the REST Adapter appears.

- c. Enter the following information:

Field	Description
Security	From the list, select OAuth 2.0 Or Basic Authentication to secure incoming messages.
Access type	Ensure Public gateway is selected. This access type uses the public internet to receive messages.

- d. Click **Test** to conform that you configured the connection successfully. When testing completes, a success message is displayed.
- e. Click **Save**.
- f. Click **Back**  to return to the integration canvas. A menu shows that the newly-created **Trigger1** connection is available for selection.



Step 4. Specify How the Trigger Connection Receives Messages in the Integration

Now let's configure the REST Adapter connection in the integration. During configuration, you specify details such as the operation, the relative resource URI, the query parameters, and the message payload content.

1. Select **Trigger1**.
The Basic Info page of the Adapter Endpoint Configuration Wizard opens. This wizard enables you to configure the **Trigger1** connection for use in the integration.
2. In the **What do you want to call your endpoint** field, enter `helloWorld`, then click **Continue** to access the Resource Configuration page.
3. Enter the following information.

Field	Description
What is the endpoint's relative resource URI	Enter the following: <code>/names/{name}</code>
What action do you want to perform on the endpoint	Select GET .

Field	Description
Select any options that you want to configure	Scroll to the bottom of the page and ensure that the following options are selected. <ul style="list-style-type: none"> • Add and review parameters for this endpoint • Configure this endpoint to receive the response

The completed Resource Configuration page looks as follows.

What does this operation do?
Describe what this operation does

What is the endpoint's relative resource URI? * ⓘ
/names/{name}

What action do you want to perform on the endpoint? * ⓘ
GET

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.
Select any options that you want to configure:

Add and review parameters for this endpoint
 Configure a request payload for this endpoint
 Configure this endpoint to receive the response

4. Click **Continue** to access the Request Parameters page.
 - a. In the **Specify Query Parameters** section, click **Add**.
 - b. Double-click the row in the **Name** column to enter a value.

<input type="checkbox"/>	Name	Data Type
<input type="checkbox"/>	email	string

* Double click to edit table cells and hit Enter/Return key to commit changes

- c. Enter `email`, then select **string** from the **Data Type** list.
 - d. Click **Add** again.
 - e. Enter `flowid`, then select **string** from the **Data Type** list.
- Both query parameters are displayed. Note also that the **name** template parameter from the relative resource URI you specified in Step 3 is also included. You select these parameters later when designing the business logic for your integration.

<input type="checkbox"/>	Name	Data Type
<input type="checkbox"/>	email	string
<input type="checkbox"/>	flowid	string

* Double click to edit table cells and hit Enter/Return key to commit changes

Template Parameters
Displays the template parameters in the relative resource URI. Template parameters are determined by details you specified on the Basic Info page and cannot be edited.

Name

name

*

string

- f. Click outside the **flowid** row.
 - g. Click **Continue** to access the Response page.
5. Enter the following information.

Field	Description
Select the response payload format	Select JSON Sample .
<<< inline >>>	Click this field and enter the following content, then click OK .
	{ "Hello" : "Name", "Message" : "Email Sent", "Email" : "EmailAddress" }
Element	Ensure response-wrapper is selected.
What is the media-type of Response Body	Ensure JSON is selected.

The completed Response page looks as follows.

Select the response payload format

JSON Sample ▾

Schema Location ⓘ

Drag and Drop
Select a file or drop one here.

--OR-- enter sample JSON
<<< inline >>>
Element

response-wrapper ▾

What is the media-type of Response Body? (Accept Header)

XML

XML(text)

JSON

6. Click **Continue** to access the Summary page.

7. Review your selections, then click **Finish**.

The REST Adapter trigger connection is now fully configured for use in the integration.

Step 5. Design How the Integration Receives Messages

Now let's fully design the business logic for your integration. You first create and design a switch action with two paths. The path taken at runtime is based on whether you specify an email address:

- Route 1: This path is taken if you specify an email address.
- Otherwise: This path is taken if you don't specify an email address.

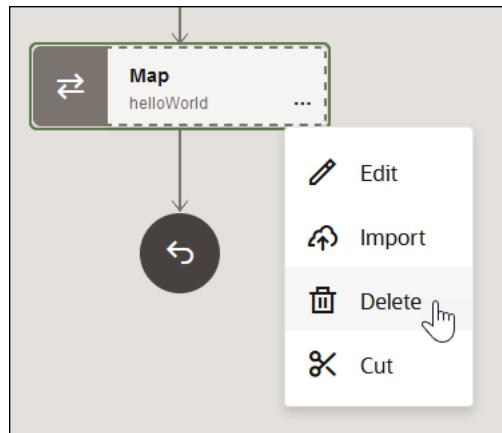
Note:

You use the Expression Builder to design much of the business logic for your integration. Be careful while entering your expressions. Simple mistakes such as forgetting a comma or entering an extra blank space can cause validation failure.

Design the Route 1 path of the integration

Let's first delete the map action that was automatically created when you configured the REST Adapter trigger connection in the integration. Instead, you add the necessary map actions for mapping source data to target data inside both switch action paths.

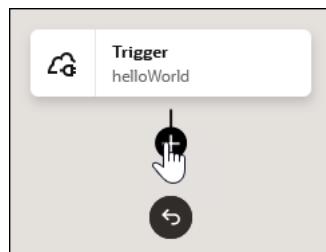
1. In the map action, click **Actions ...**, then select **Delete**.



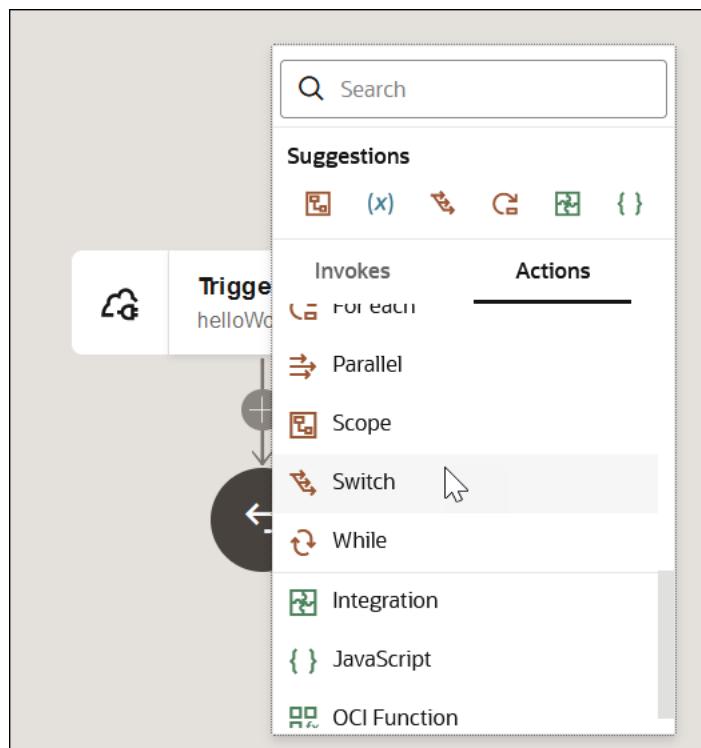
2. Click **Delete** when prompted.

Now let's add a switch action to define routing expressions in your integration. The switch action takes the first path that evaluates to true. The other paths are ignored.

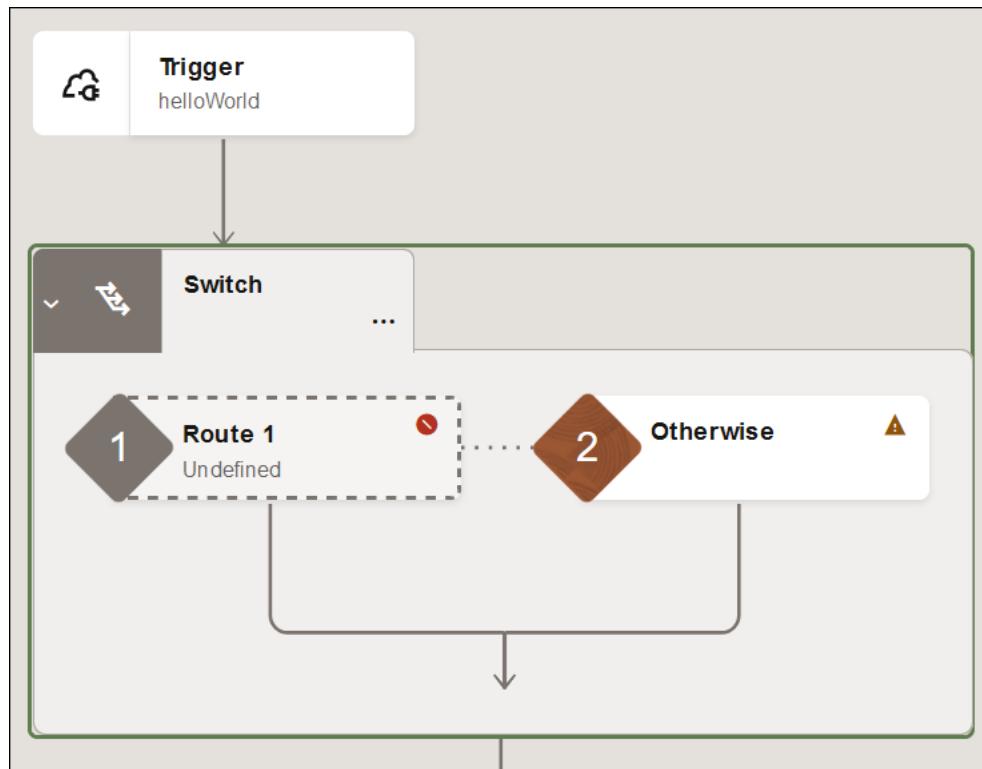
3. Hover over the line below the trigger connection and click **Add**



4. Click **Actions**, then **Switch**.



- In the switch action, click **Actions ...**, then select **Add**, and then **Otherwise**. A switch action with two branches is displayed.



- Hover your cursor inside **Route 1**, and select **Actions ...**, then **Edit** . The Configure route panel opens.

- Click **Edit** .



- Enter `sendEmail` as the name for the switch action, then click **Apply** .

- To the right of the **Value** field, click **Switch to Developer View** .
- In the **Sources** tree, expand **QueryParameters**. These are the query parameters you defined when configuring the REST Adapter in the integration.
- Drag **email** to the **Value** field. This parameter lets you specify the email address that gets logged to the activity stream when you test and run the integration.

The screenshot shows the Oracle Integration Designer's Expression Builder. On the left, there's a sidebar with 'Input sources' and 'Functions' tabs. Under 'Sources', there are sections for 'execute *', 'TemplateParameters *', 'QueryParameters', and 'ConnectivityProperties *'. 'QueryParameters' is expanded, showing 'email' (selected), 'flowid', and 'ConnectivityProperties *'. In the main panel, the routing expression is defined as ': Configure route sendEmail'. Below it is the expression 'email != '''. To the right of the expression, there are two error messages: 'Invalid Statement' and another 'Invalid Statement' with a 'Value' field containing '/ns12:execute /ns12:QueryParameters/ns17:email'. A small error icon is visible on the right side of the main panel.

- g. In the **Operator** list, select **!=**.
- h. In the **Value** field, enter two single quotes '' with *no* blank space in between.
The completed routing expression looks as follows. When the email address that you specify at runtime is not an empty value, this route of the switch action is taken.

The screenshot shows the 'Configure route' panel with the completed routing expression: 'email != '''. An error icon is visible on the right side of the panel.

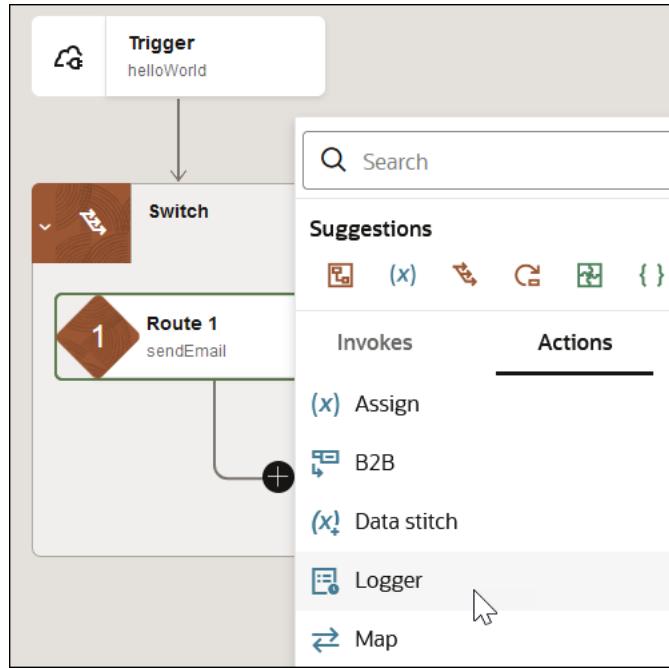
Note:

As you design this integration, note that an error icon can appear on the right side of the page. Click this icon for details. Errors may occur due to mistakes you make in the Expression Builder. Follow the instructions in the Error panel to resolve the errors. Other errors that appear are resolved as you complete the design of all parts of this integration. Click outside the error panel to close it.

- i. Click **Save**, then click outside the Configure route panel to close it.

Now let's add a logger action. This action enables you to log messages to the activity stream at runtime for diagnostic purposes.

6. Drag your cursor along the line below **Route 1** until the **Add** icon appears.
7. Click **Add** , then select **Actions**, then **Logger**.



8. The Configure logger panel appears. The **Input Sources** and **Functions** tabs appear in the upper left part of the panel. You use the elements available under these tabs to build a logger expression.

- a. Click **Edit**.
- b. Enter `logincoming` as the name for the logger action, then click **Apply ✓**.
- c. To the right of the **Logger message** field, click **Switch to Developer Mode**.
- d. Click the **Functions** tab, then expand **String**.
- e. Drag a **concat** function into the **Logger message** field.

Input sources	Functions	Configure logger
		logincoming
Components		Log messages to the activity stream and diagnostic logs by adding the logger action at any point in the integration. You can create a log message that is a static message or variable-populated message in the expression builder.
Functions	<ul style="list-style-type: none"> > Node-set ▼ String <ul style="list-style-type: none"> <code>compare</code> <code>compare-ignore-case</code> concat <code>contains</code> 	<p>Log *</p> <p><input checked="" type="radio"/> Always</p> <p><input type="radio"/> When tracing level is set to Audit</p> <p>(x) Logger message * <code>fn:concat(string1, string2, ...)</code></p> <p><small>Typing is only available for literals and numbers in designer view. You can also toggle to developer view for full editing capabilities.</small></p>

- f. Remove the content between the parentheses.

- g. After the opening parenthesis, enter the following:

```
'Hello World was invoked by ',
```

Ensure that you enter a blank space before the single quote and after the comma.

- h. Click the **Input sources** tab, and expand **TemplateParameters**.
i. Drag **name** before the closing parenthesis.

Input sources	Functions
Sources	Configure logger logincoming Log messages to the activity stream and diagnostic logs by adding the logger action at any point in the integration. You can create a log message that is a static message or variable-populated message in the expression builder. <input checked="" type="radio"/> Always <input type="radio"/> When tracing level is set to Audit Logger message * <code>fn:concat('Hello World was invoked by ', /ns12:execute/ns12:TemplateParameters /ns17:name)</code>

- j. Before the closing parenthesis, enter the following:

```
, '. Email will be sent to ',
```

Note the single blank space after the last comma.

- k. Expand **QueryParameters**.
l. Before the closing parenthesis, drag **email**.

Input sources	Functions
Sources	Configure logger logincoming Log messages to the activity stream and diagnostic logs by adding the logger action at any point in the integration. You can create a log message that is a static message or variable-populated message in the expression builder. <input checked="" type="radio"/> Always <input type="radio"/> When tracing level is set to Audit Logger message * <code>fn:concat('Hello World was invoked by ', /ns12:execute/ns12:TemplateParameters /ns17:name, '. Email will be sent to ', /ns12:execute/ns12:QueryParameters /ns17:email)</code>

- m. Before the closing parenthesis, enter the following:

```
, '!!'
```

The completed logger action looks as follows.

The screenshot shows the 'Configure logger' panel for an action named 'logincoming'. The panel includes a description of logger actions, a 'Log' section with a radio button for 'Always', and a code editor containing the following expression:

```
(x) Logger message*
fn:concat('Hello World was invoked by ',
/ns12:execute/ns12:TemplateParameters
/ns18:name, '. Email will be sent to ',
/ns12:execute/ns12:QueryParameters
/ns18:email, '!')
```

There are edit and delete icons next to the code editor.

- n. Click **Switch to Designer View**  to toggle to designer view, which shows a more user-friendly version of the expression.

The screenshot shows the same logger configuration in designer view. The expression is now displayed in a simplified, readable form:

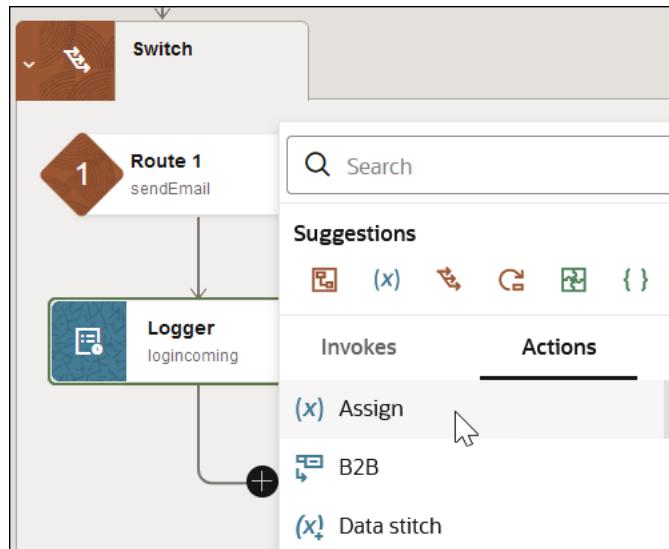
```
(x) Logger message*
concat('Hello World was invoked by ', name, '.'
Email will be sent to ', email, '!')
```

A tooltip at the bottom left indicates: "Typing is only available for literals and numbers in designer view. You can also toggle to developer view for full editing".

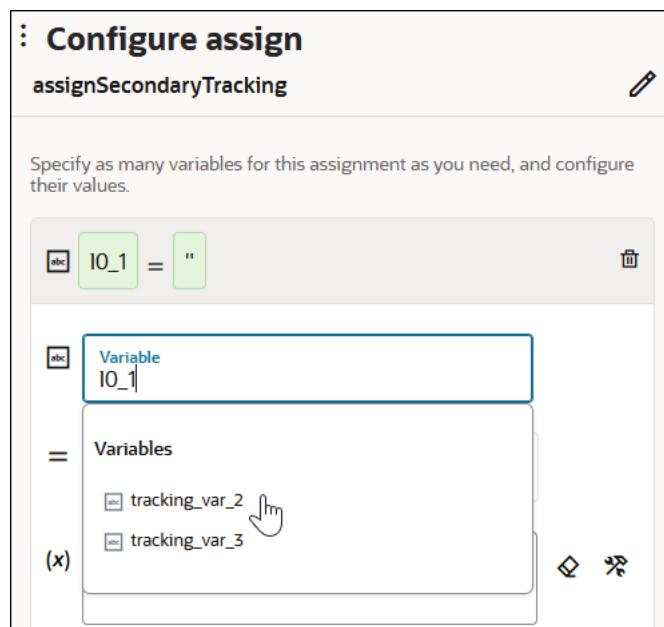
- o. Click **Save**, then click outside the Configure logger panel to close it.

Now let's assign an optional secondary business identifier to track payload fields in messages during runtime. Each integration requires at least one business identifier (known as the primary business identifier) to track the status of your running instance (for example, was the run successful or did the run fail). Later in this tutorial, you assign the primary business identifier from a separate location in the integration canvas.

9. Drag your cursor along the line below **Logger** until the **Add**  icon appears.
10. Click **Add** , then select **Actions**, then **Assign**.



- Click **Edit**.
- Enter `assignSecondaryTracking` as the name for the assign action, then click **Apply** ✓.
- Click **Add**.
- Click inside the **Variable** field, and select `tracking_var_2` from the list.



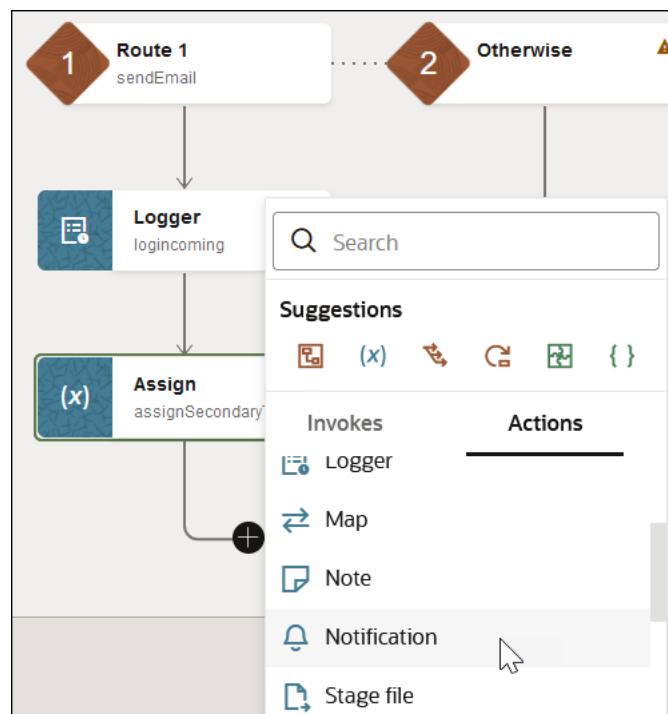
- To the right of the **Value** field, click **Switch to Developer View**.
- Clear the content from the **Value** field.
- Expand **QueryParameters**, and drag **email** into the **Value** field.

The screenshot shows the Oracle Integration Designer's configuration interface. On the left, there's a sidebar with 'Input sources' and 'Functions' tabs. Under 'Sources', there are several expandable sections: 'execute *', 'TemplateParameters *', 'QueryParameters', 'email' (which is selected and highlighted in blue), 'flowid', 'ConnectivityProperties *', '\$self', '\$tracking_var_1', '\$tracking_var_2', and '\$tracking_var_3'. To the right, a panel titled 'Configure assign' is open, showing a configuration for 'assignSecondaryTracking'. It includes a condition 'tracking_var_2 = email', an operation 'Assign', and a value path '/ns12:execute/ns12:QueryParameters/ns17:email'. A red box highlights the value path.

- Click **Save**, then click outside the Configure assign panel to close it.

Now let's design the contents of the notification email you receive after running the integration.

- Drag your cursor along the line below **Assign** until the **Add** icon appears.
- Click **Add** , then select **Actions**, then **Notification**.

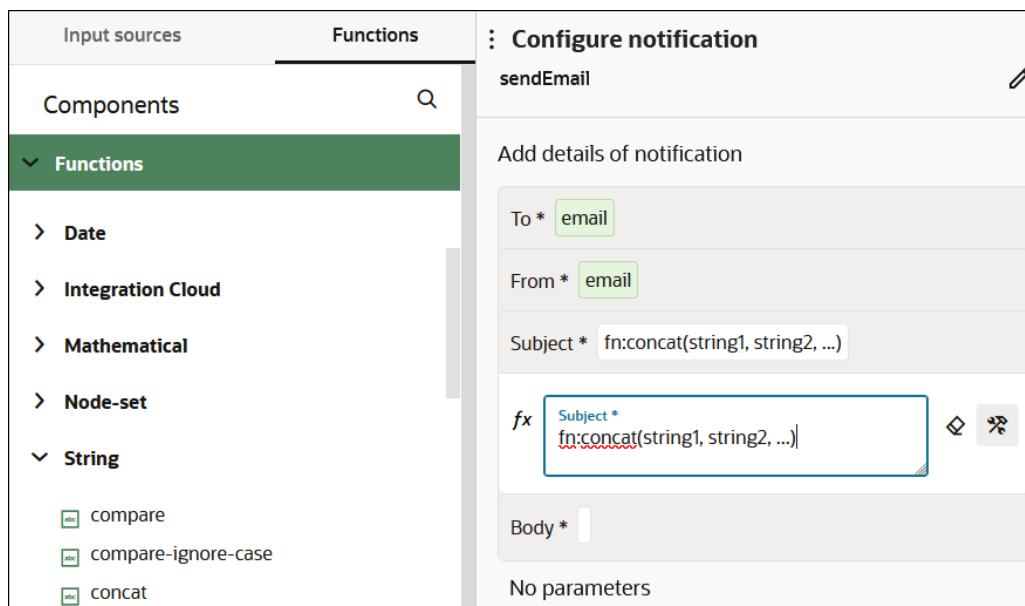


The Configure notification panel opens.

- Click **Edit** .

- b. Enter `sendEmail` as the name for the notification action, then click **Apply ✓**.
- c. Click the **To** field.
The **Sources** panel opens.

- d. Click **Switch to Developer View** .
- e. Expand **QueryParameters**, and drag **email** into the **To** field.
- f. Click the **From** field.
- g. Expand **QueryParameters**, and drag **email** into the **From** field.
- h. Click **Switch to Developer View**  again.
- i. Click the **Subject** field.
- j. Click the **Functions** tab, then expand **String**.
- k. Drag **concat** into the **Subject** field.



The screenshot shows the Oracle Notification configuration interface. On the left, there's a sidebar with 'Input sources' and 'Components' sections. The main area has a title 'Configure notification' with a sub-section for 'sendEmail'. It includes fields for 'To', 'From', and 'Subject'. The 'Subject' field contains the function `fn:concat(string1, string2, ...)`. A tooltip for this field shows the expanded function `fn:concat(string1, string2, ...)`. Below the fields, it says 'No parameters'.

- l. Remove the content between the parentheses.
- m. After the opening parenthesis, enter the following:

`'Hello' ,`

Note the blank space after the comma.

: Configure notification

sendEmail

Add details of notification

To * email

From * email

Subject * fn:concat('Hello',)

(x) Subject * fn:concat('Hello',)

Please provide a valid expression by selecting an element from the dropdown, or drag an element from the source tree.

- n. Click the **Input sources** tab, expand **TemplateParameters**, and drag **name** into the **concat** function before the closing parenthesis.

Input sources	Functions
Sources	
▼ execute *	
▼ Ⓛ TemplateParameters *	
☒ name *	
▶ Ⓛ QueryParameters	
▶ Ⓛ ConnectivityProperties *	
▶ \$self	
(x) \$tracking_var_1	
(x) \$tracking_var_2	

: Configure notification

sendEmail

Add details of notification

To * email

From * email

Subject * concat('Hello' , name)

(x) Subject * fn:concat('Hello', /ns12:execute /ns12:TemplateParameters/ns17:name)

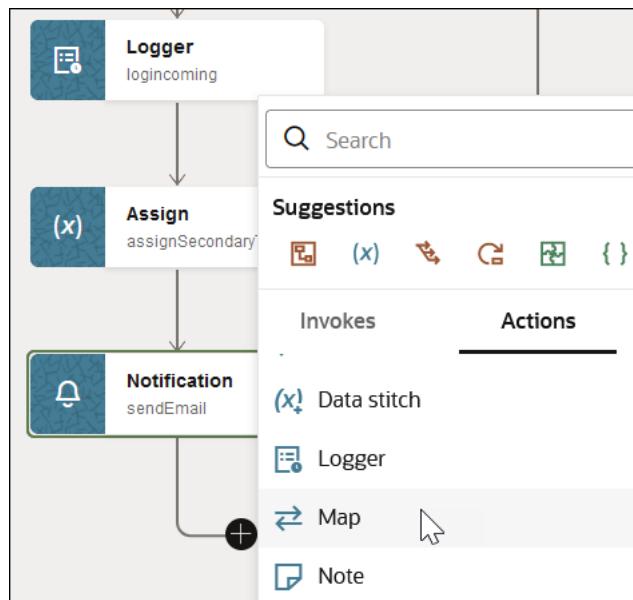
- o. Click the **Body** field and enter the following text. This is the email message you receive when the integration runs successfully.

```
<html><p><span style="color: #298022;"><strong>Welcome to Oracle Integration!</strong></span></p></html>
```

- p. Click **+** in the **No parameters** section at the bottom.
- q. In the **Parameter** field, enter **name**.
- r. Click the **Input sources** tab, expand **TemplateParameters**, and drag **name** into the **Parameter value** field.

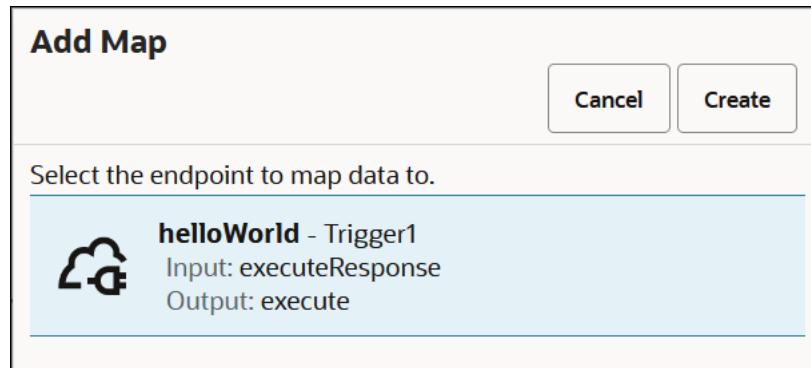
The screenshot shows the Oracle Integration Designer interface. The left panel displays a tree structure under 'Sources' with items like 'execute*', 'TemplateParameters*', '\$self', and tracking variables. A context menu is open over the 'TemplateParameters*' node, with the option 'Add details of notification' highlighted. The right panel shows a configuration dialog for a notification. It includes fields for 'To * email', 'From * email', 'Subject * concat ('Hello', name)', and 'Body * <html><p>Welcome to Oracle Integration!</p></html>'. Below these are sections for 'Parameters' and a preview of the parameter value as XML.

- s. Click **Save**, then click outside the Configure notification panel to close it.
13. Drag your cursor along the line below **Notification** until the **Add** icon appears.
14. Click **Add** , then select **Actions**, then **Map**.



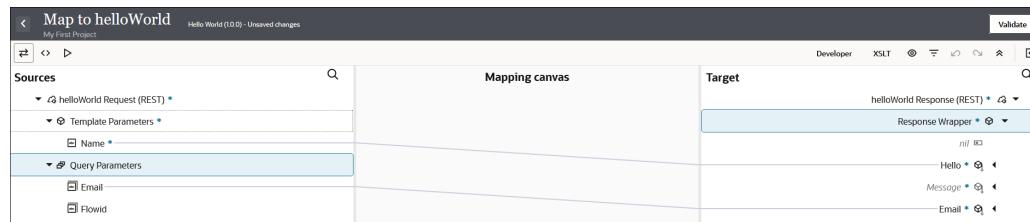
The Add Map panel opens.

- a. Select the **helloWorld** endpoint to which to map data, and click **Create**.

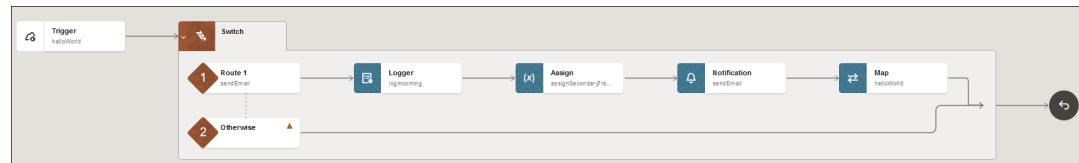


The mapper opens.

- b. Expand **Template Parameters** and **Query Parameters** in the **Sources** section and **Response Wrapper** in the **Target** section.
- c. Perform the following mappings:
 - Drag the source **Name** element to the target **Hello** element.
 - Drag the source **Email** element to the target **Email** element.



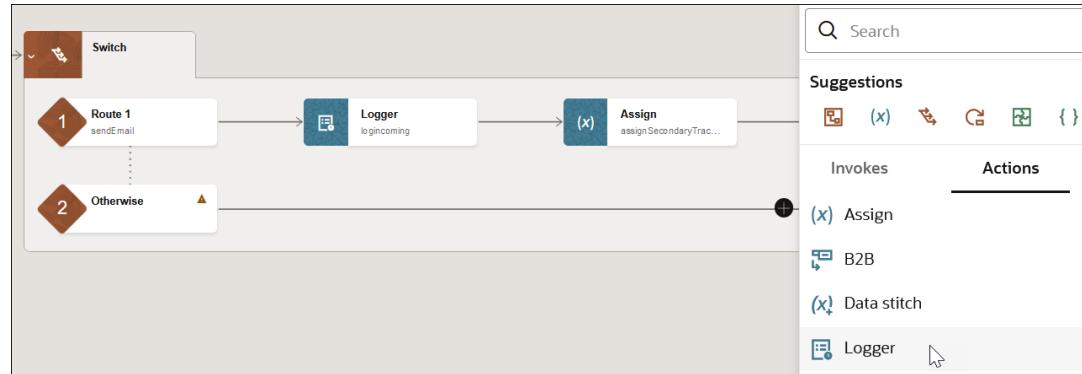
- d. Click **Validate**.
 - e. Click **Back** to exit the mapper and return to the integration canvas. Your mappings are automatically saved.
15. Above the integration, click **Horizontal** to see a horizontal view of the integration. You have now completed configuration of the first route of the switch action.



Design the Otherwise path of the integration

Now let's design the second (Otherwise) path of the integration.

1. Drag your cursor along the line to the right of **Otherwise** until the **Add** icon appears.
2. Click **Add** , then select **Actions**, then **Logger**.



- Click **Edit** .
- Enter `logIncomingWithoutEmail` as the name for the logger action, then click **Apply** .
- In the **Logger message** field, click **Switch to Developer Mode** .
- Click the **Functions** tab, then expand **String**.
- Drag a **concat** function into the **Logger message** field.
- Remove the content between the parentheses.
- After the opening parenthesis, enter the following:

`'Hello World was invoked by ',`

Note the single blank space after the comma.

: Configure logger

logIncomingWithoutEmail 

Log messages to the activity stream and diagnostic logs by adding the logger action at any point in the integration. You can create a log message that is a static message or variable-populated message in the expression builder.

Log *

Always

When tracing level is set to Audit

Logger message *

`fn:concat('Hello World was invoked by ',)`

- Click the **Input sources** tab, and expand **TemplateParameters**.
- Drag **name** before the closing parenthesis.

Configure logger

logIncomingWithoutEmail

Log messages to the activity stream and diagnostic logs by adding the logger action at any point in the integration. You can create a log message that is a static message or variable-populated message in the expression builder.

Log *

Always

When tracing level is set to Audit

Logger message *

```
fn:concat('Hello World was invoked by ', /ns12:execute/ns12:TemplateParameters /ns17:name, ')
```

- j. Before the closing parenthesis, enter the following:

Configure logger

logIncomingWithoutEmail

Log messages to the activity stream and diagnostic logs by adding the logger action at any point in the integration. You can create a log message that is a static message or variable-populated message in the expression builder.

Log *

Always

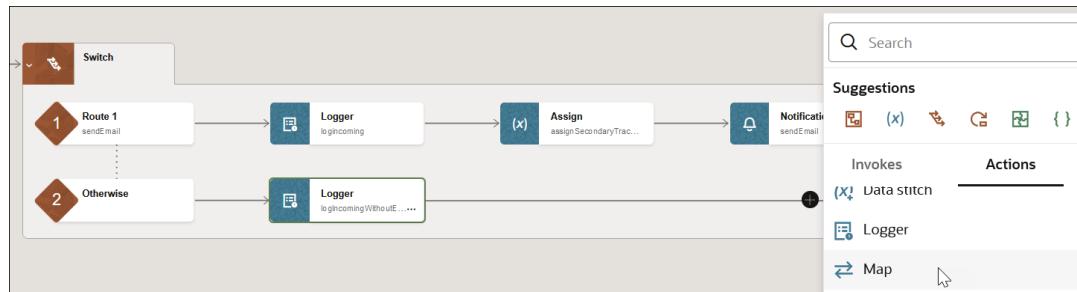
When tracing level is set to Audit

Logger message *

```
fn:concat('Hello World was invoked by ', /ns12:execute/ns12:TemplateParameters /ns18:name, ')
```

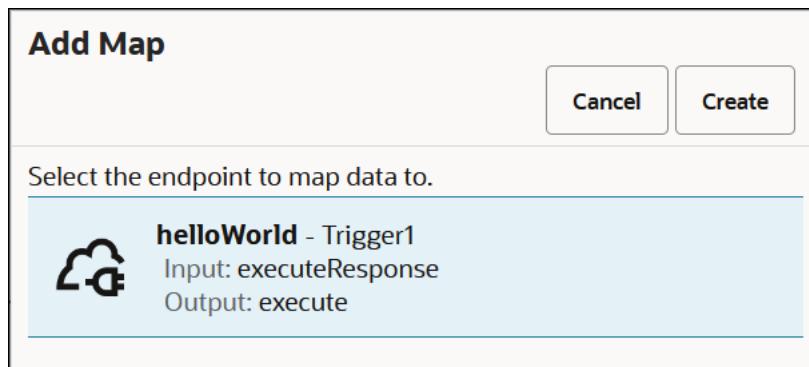
- k. Click **Save**, then click outside the Configure logger panel to close it.

3. Drag your cursor along the line to the right of the **Logger** until the **Add**  icon appears.
4. Click **Add** , then select **Actions**, then **Map**.



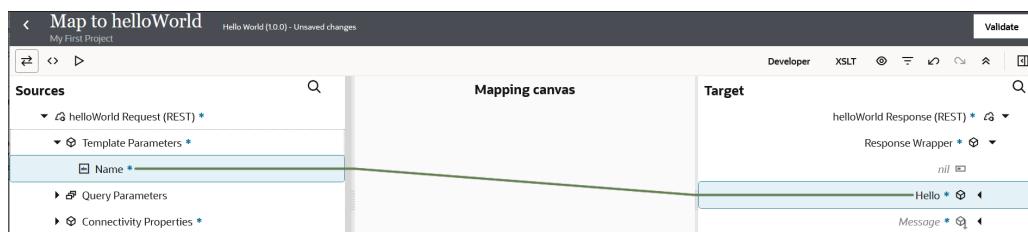
The Add Map panel opens.

- Select the **helloWorld** endpoint to which to map data and click **Create**.



The mapper opens.

- Expand **Template Parameters** in the **Sources** section and **Response Wrapper** in the **Target** section.
- Drag the source **Name** element to the target **Hello** element.



- Click **Validate**.
- Click **Back** ↺ to exit the mapper and return to the integration canvas. Your mappings are automatically saved.

Both routes are now configured.



Step 6. Define a Business Identifier to Track Message Delivery at Runtime

Business identifiers enable you to track payload fields in messages during runtime. Each integration requires a primary business identifier. Without specifying a business identifier, you cannot complete design integration or run your integration.

1. Above the integration, click **Business Identifier (I)**.
- a. Expand **TemplateParameters**, and drag **name** to the **Business identifier** field.

- b. Click **Save**, then click outside the Business identifiers panel to close it.
2. Check for any red error icons. If you see any, click the icon for details and resolve those errors.
3. Click **Back** to exit the integration canvas and return to the project page.

Step 7. Activate and Run the Integration

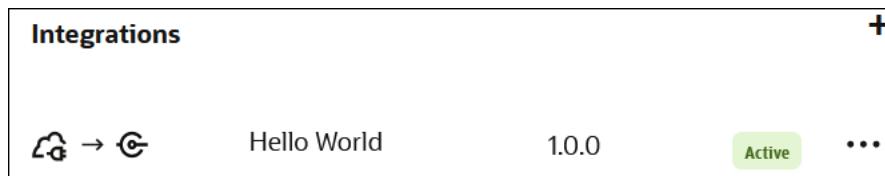
You are now ready to activate the integration to the runtime environment. Once activated, you can run the integration.

1. In the **Integrations** section, click **Actions ...**, then select **Activate**.

Icon	Name	Version	Status
	Hello World	1.0.0	Configured

2. Keep the tracing level set to **Production**.
3. Click **Activate**.

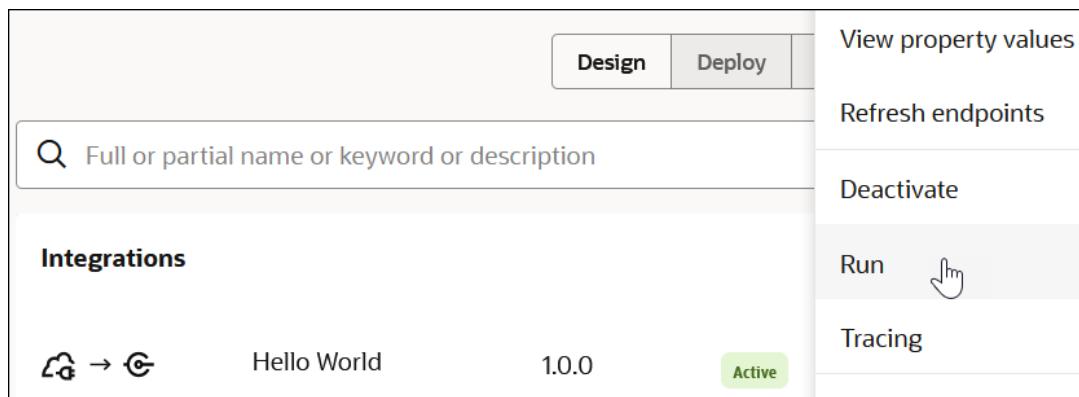
4. Click **Refresh**  periodically.
The integration status changes to **Active**.



The screenshot shows the Oracle Integration console interface. At the top, there's a header with the title 'Integrations'. Below it, a card displays the integration details: 'Hello World', version '1.0.0', and a green button labeled 'Active'. To the right of the card is a vertical ellipsis menu.

You are now ready to run the integration and send a message.

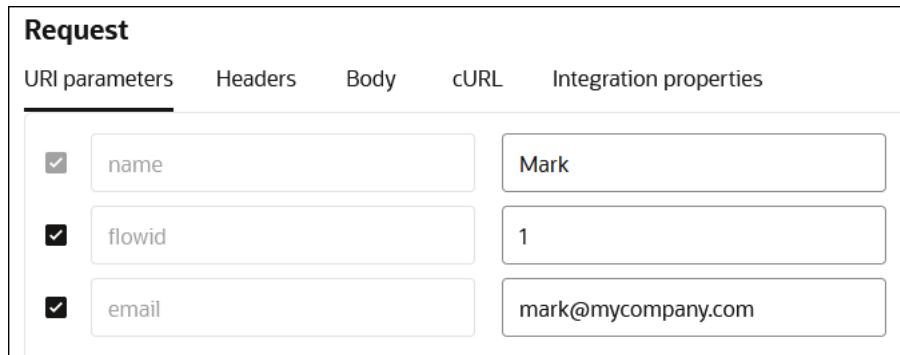
5. In the **Integrations** section, click **Actions** ..., then select **Run**.



The screenshot shows a context menu for the 'Hello World' integration. The 'Run' option is highlighted with a mouse cursor icon. Other options visible in the menu include 'View property values', 'Refresh endpoints', 'Deactivate', and 'Tracing'.

The Configure and run page enables you to run and test REST Adapter trigger-based integrations. This page provides a way of testing the sending of an inbound message from an actual application.

6. In the **Request** section, specify your name, a flow ID number, and your email address.



The screenshot shows the 'Request' configuration page. Under the 'URI parameters' tab, three fields are defined: 'name' (value: 'Mark'), 'flowid' (value: '1'), and 'email' (value: 'mark@mycompany.com'). Each field has a checked checkbox next to it.

7. Click **Run** to send your inbound message to the integration.

8. Click **Refresh**  periodically.

The Activity stream panel opens and shows the movement of the message through the actions that you designed (switch, logger, mapper, and others). The unique instance ID that was created for this integration run is displayed at the top. All milestones in this activity stream are green, indicating the successful delivery of this message.

Activity stream

Instance Id: _YEgJQpVEe6x5b1AKrobDQ [Copy](#)

Tracing level: **Production** [Change](#) [Download](#)

Tue Jun 13 06:51:33.666 PM PDT 2023, 40ms [View](#) Wire Message received by Trigger helloWorld

06:51:33.706 PM, 1ms [View](#) Message started processing

06:51:33.707 PM [View](#) Message entered Switch

06:51:33.707 PM [View](#) Message routed through Switch Route with condition email != "

② 06:51:33.707 PM [View](#) Logger logincoming

06:51:33.707 PM, 1ms [View](#) Message received by Assignment assignSecondaryTracking

06:51:33.708 PM [View](#) Message received by Notification Notification1

② 06:51:40.930 PM [View](#) Message sent by Notification

06:51:33.725 PM, 4ms [View](#) Data Mapping completed

06:51:33.729 PM [View](#) Message exited Switch

06:51:33.729 PM, 3ms [View](#) Message sent to reply

06:51:33.732 PM [View](#) Reply to helloWorld completed

9. Click **View**  next to the logger and notification actions to view the message payload that passed through the integration.

The screenshot shows the Activity Stream panel with the following log entries:

- 08:14:53.103 PM, 1ms: Logger logincoming
- 08:14:53.104 PM: Message received by Assignment assignSecondaryTracking
- 08:14:53.104 PM: Message received by Notification Notification1
- 08:14:56.255 PM: Message sent by Notification

Below the log entries, there is a JSON representation of the email message sent:

```
{"subject": "Hello Mark", "fromAddress": "mark.kennedy@oracle.com", "toAddress": "mark.kennedy@oracle.com", "mimeType": "text/html", "body": "<p><span style=\"color: #298022;\"><strong>Welcome to Oracle Integration!</strong></span></p></html>"}
```

Step 8. Monitor Message Delivery Status at Runtime

- Close the Activity stream panel and click **Track instances**.

The **Observe** tab in your project opens. This tab enables you to monitor the status of your integration instance. The status of the instance is displayed as **Succeeded**.

The screenshot shows the Observe tab with the following details:

- Time window:** Last 1 hour
- Integration:** Hello World (HELLO_WORLD | 01.00.0000)
- Status:** Succeeded
- Duration:** 2 minutes ago
- Business Identifiers:** tracking..., mark.kennedy@oracle.com

Primary Identifier	Instance Id	Business Identifiers	Status	Duration
name: Mark Hello World 1.0.0	9jkyKAvzEe6w7mfP4...	tracking..., mark.kennedy@oracle.com	Succeeded	2 minutes ago 2 minutes ago 83 milliseconds

Legend: Succeeded (Green), In progress (Yellow), Blocked (Red), Encoded (Blue), Queued (Purple), Resubmitted (Grey), Aborted (Teal), Passed (Light Green).

- Check your email inbox for a new message.
- Open the email and note the following message:

The email message content is:

[External] : Hello Mark

To: Mark

Welcome to Oracle Integration!

Congratulations! You have successfully created, designed, activated, run, and monitored your first integration. See the sections below for more information about the capabilities used in this tutorial.

Learn More About The Features in this Tutorial

- [About Projects](#)
- [About Integrations](#)
- [About Connections](#)
- [Add Actions to an Application Integration](#)
- [Assign Business Identifiers](#)
- [Activate an Integration](#)
- [Test Integrations from Outside the Integration Canvas](#)
- [Monitor Integrations in a Project](#)

Tutorial: Build Your Second Integration From Scratch

Let's walk through how to create, design, activate, run, and monitor a second integration completely from scratch. For this tutorial, you create an integration that invokes a weather application to return a weather report for a city that you specify.

- [What Do I Learn with this Tutorial](#)
- [Step 1. Complete Prerequisites](#)
- [\(Optional\) Step 2. Test the API Key Outside of Oracle Integration](#)
- [Step 3. Create a Project](#)
- [Step 4. Create an Integration](#)
- [Step 5. Create a Trigger Connection to Define a City Parameter and the Content of the Weather Report Response](#)
- [Step 6. Create an Invoke Connection to Call the OpenWeather Application](#)
- [Step 7. Specify How the Invoke Connection Retrieves the Latitude and Longitude Coordinates for the City](#)
- [Step 8. Specify How the Invoke Connection Retrieves the Weather Report for the City](#)
- [Step 9. Map the Weather Report Response Returned from the OpenWeather Application](#)
- [Step 10. Define a Business Identifier](#)
- [Step 11. Activate and Run the Integration](#)
- [Step 12. Monitor the Integration](#)
- [Learn More About The Features in this Tutorial](#)

What Do I Learn with this Tutorial

This tutorial walks you through performing the following tasks:

- Creating a project in which to perform all design, management, and monitoring tasks.
- Creating an integration in which to design your business requirements.

- Creating and configuring a trigger connection in which you define a city parameter and the contents of the weather report to receive back from the OpenWeather application (city, latitude, longitude, and other content).
- Creating and configuring an invoke connection to obtain the latitude and longitude coordinates for the specified city from the OpenWeather application.
- Creating and configuring a second invoke connection to obtain the weather report from the OpenWeather application.
- Specifying a city to start the integration and receiving the weather report back for that city from the OpenWeather application.
- Monitoring the results of the integration run.

Step 1. Complete Prerequisites

When you configure the REST Adapter as an invoke connection to use in this integration, you select the API Key Based Authentication security policy. This policy requires you to specify an API key. Create an account at the OpenWeather application to obtain an API key.

1. Go to the [OpenWeather Application](#).
2. Create an account to use the application.
3. Sign in by specifying your email address and password.
4. Click **API keys** or click your profile link in the upper right corner.
5. Copy and save your API key value.

(Optional) Step 2. Test the API Key Outside of Oracle Integration

When you test REST Adapter configuration later in this tutorial, a message appears indicating that the API key is not validated as part of the test. You must validate the API key separately. While there may be nothing to indicate that your API key value is invalid, it is typically a best practice to test your connection outside of Oracle Integration to completely validate the API key.

You can use a third-party tool such as Postman to test your API key connection outside of Oracle Integration. This test provides a way to completely validate the API key. For example:

```
https://api.openweathermap.org/?appid=API_key_value&units=imperial
```

Where *API_key_value* is the key you obtained from the OpenWeather application.

For example, from a third-party tool such as Postman, you specify and test the URL and query parameter values for `appid` and `units`. Because this tutorial returns the temperature in degrees Fahrenheit, `imperial` is specified. You can also specify `metric` to return the temperature in degrees Celsius.

The screenshot shows a REST client interface with a 'GET' method selected. The URL is set to <https://api.openweathermap.org/>. In the 'Params' tab, there are two query parameters: 'appid' with value 'API_key_value' and 'units' with value 'imperial'. A 'Send' button is visible at the top right.

Step 3. Create a Project

You manage the entire life cycle of an integration from within a project.

1. In the navigation pane, click **Projects**.
2. Click **Add**, then select **Create**.
3. In the **Name** field, enter `My Weather Project`.
4. Leave all remaining fields as they are, then click **Create**.
The page for your project is displayed. Your project contains sections for creating, designing, deploying, and monitoring your integration resources (integrations, connections, lookups, and JavaScript libraries).

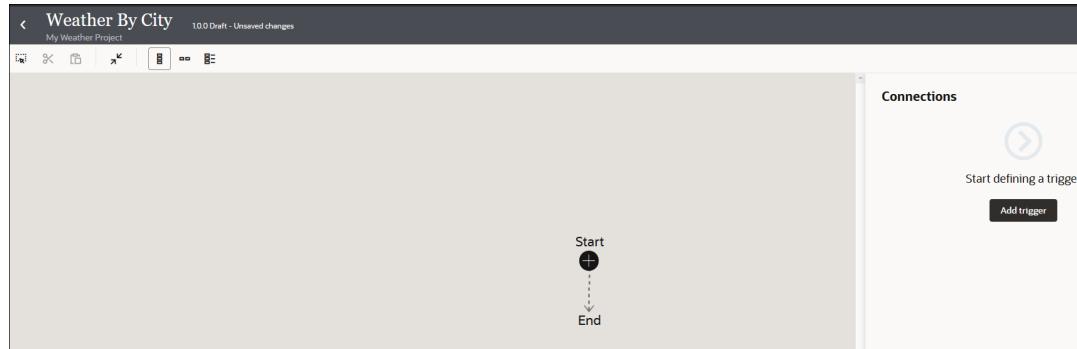
Step 4. Create an Integration

Let's now create an integration inside the project.

1. In the **Integrations** section, click **Add**.
The Add integration panel opens.
2. Click **Create**, then select **Application**.

The screenshot shows the 'Create integration' dialog. It has two main options: 'Application' and 'Schedule'. The 'Application' option is selected and described as 'Starts running when it receives an event from an application'. The 'Schedule' option is also listed.

3. In the **Name** field, enter `Weather By City`.
4. Leave all remaining fields as they are, then click **Create**.
The integration canvas opens with an option for creating a trigger connection. The trigger connection enables your integration to receive inbound messages from an application.



 **Note:**

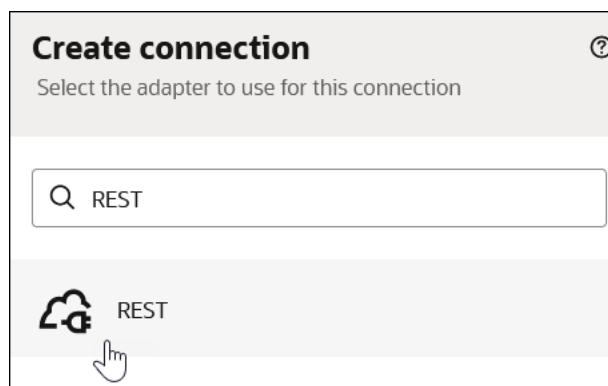
As you design this integration, an error icon can appear on the right side of the page. Errors may occur due to mistakes you make in the mapper or Adapter Endpoint Configuration Wizard. Other errors that appear are resolved as you complete the design of all parts of this integration. Click the error icon to view details. To resolve the error, you can either follow the instructions in the Error panel or wait to see if your completed design resolves the issues. Click outside the Error panel to close it.



Step 5. Create a Trigger Connection to Define a City Parameter and the Content of the Weather Report Response

Let's now create a trigger connection inside the integration canvas. The trigger (source) connection allows you to define a city parameter and the content of the weather report to receive back from the OpenWeather application (city, latitude, longitude, and other content).

1. In the Connections panel of the integration canvas, click **Add trigger**.
The Create connection panel opens.
2. Enter REST in the search field.
3. Select **REST** as the adapter connection to use. For this tutorial, the REST Adapter processes inbound messages sent to Oracle Integration.



4. The page for configuring the REST Adapter opens.

- Enter the following information.

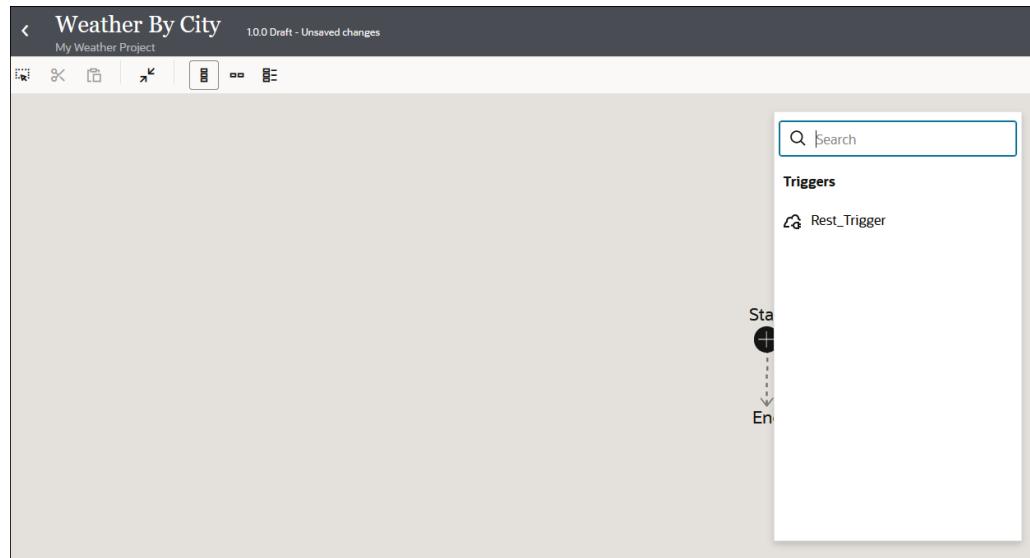
Field	Description
Name	Enter Rest_Trigger.
Role	<p>Select Trigger. Roles perform the following tasks.</p> <ul style="list-style-type: none"> Trigger: Enables an inbound application (endpoint) to trigger (run) the integration. Invoke: Enables your integration to invoke an outbound application (endpoint). Trigger and Invoke: Enables you to create a connection that can be used as both a trigger and invoke. For example, you use the connection as a trigger in one integration and as an invoke in another integration.

- Leave all remaining fields as they are, then click **Create**.
The page for configuring security and delivery methods for the REST Adapter trigger connection appears.
- Enter the following information:

Field	Description
Security	Select OAuth 2.0 to secure incoming messages.
Access type	Ensure Public gateway is selected. This access type uses the public internet to receive messages.

- Click **Test** to confirm that you configured the connection successfully. When testing completes, a success message is displayed. Configuration progress is now 100% complete.

- Click **Save**.
- Click **Back** to return to the integration canvas. A menu shows that the newly-created **Rest_Trigger** connection is available for selection.



Let's now configure the REST Adapter trigger connection in the integration. During configuration, you specify details such as the operation, the relative resource URI, the query parameters, and the message payload content.

5. In the integration canvas, click the **Rest_Trigger** connection shown in the menu. The Basic Info page of the Adapter Endpoint Configuration Wizard opens. This wizard enables you to configure the **Rest_Trigger** connection for use in the integration.
6. In the **What do you want to call your endpoint** field, enter `GetWeatherByCity`, then click **Continue** to access the Resource Configuration page.
7. Enter the following information.

Field	Description
What is the endpoint's relative resource URI	Enter the following: /weather
What action do you want to perform on the endpoint	Select GET .
Select any options that you want to configure	Scroll to the bottom of the page and select the following options: <ul style="list-style-type: none">• Add and review parameters for this endpoint• Configure this endpoint to receive the response

8. Leave all remaining fields as they are.
The completed Resource Configuration page looks as follows.

What is the endpoint's relative resource URI? * [?](#)

/weather

What action do you want to perform on the endpoint? * [?](#)

GET

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.

Select any options that you want to configure:

- Add and review parameters for this endpoint
- Configure a request payload for this endpoint
- Configure this endpoint to receive the response

9. Click **Continue** to access the Request Parameters page.
 - a. In the **Specify Query Parameters** section, click **Add**.
 - b. Double-click the row in the **Name** column to enter a value.

<input type="checkbox"/>	Name	Data Type
<input type="checkbox"/>		string

* Double click to edit table cells and hit Enter/Return key to commit changes

- c. Enter `city`, then select **string** from the **Data Type** list.

Specify Query Parameters

Add Remove

<input type="checkbox"/>	Name	Data Type
<input checked="" type="checkbox"/>	city	string

At runtime, you specify the city on the Configure and run page to receive a weather report.

- d. Click outside the `city` row.
- e. Click **Continue** to access the Response page.
10. Enter the following information.

Field	Description
Select the response payload format	Select JSON Sample .

Field	Description
<<< inline >>>	Click this field and enter the following content, then click OK .
	{ "city" : "Timbuktu", "longitude" : 1.1, "latitude" : 1.2, "country" : "AF", "description" : "clear sky", "temp" : 20.6 }
Element	Ensure response-wrapper is selected.
What is the media-type of Response Body	Ensure JSON is selected.

The completed Response page looks as follows.

Select the response payload format

Sample Location [?](#)

Drag and Drop
Select a file or drop one here.

--OR-- enter sample JSON
<<< inline >>>
Element

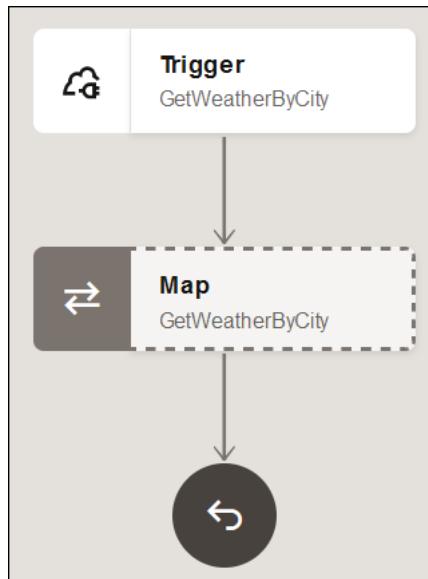
What is the media-type of Response Body? (Accept Header)

XML
 XML(text)
 JSON

11. Click **Continue** to access the Summary page.

12. Review your selections, then click **Finish**.

The REST Adapter trigger connection is now fully configured for use in the integration. A map action is also automatically created for mapping data between the trigger connection and the invoke connection that you create in the next section.



Step 6. Create an Invoke Connection to Call the OpenWeather Application

Let's now create an invoke connection inside the integration canvas to call the OpenWeather application and receive weather information.

1. In the right pane, click **Invokes** . The Create connection panel opens.
2. Click **Create Connection** to create a new connection.



3. Select **REST** as the adapter connection to use. For this tutorial, the REST Adapter invokes the OpenWeather application.
 - a. Enter the following information.

Field	Description
Name	Enter Weather.
Role	Select Invoke . This role enables your integration to invoke the OpenWeather application.

- b. Leave all remaining fields as they are, then click **Create**. The page for configuring security and delivery methods for the REST Adapter invoke connection appears.
- c. Enter the following information:

Field	Description
Connection Type	Select REST API Base URL to secure the message.
Connection URL	Enter the following. <code>https://api.openweathermap.org/</code>
Security policy	Select API Key Based Authentication . This is the authentication security used by the OpenWeather application.
API Key	Enter the API key you obtained in Step 1. Complete Prerequisites .
API Key Usage	Expand Optional security , then enter one of the following key usages. This is how the key is passed to the OpenWeather application. The value of <code>imperial</code> indicates that we want to return the temperature in degrees Fahrenheit. <code>?appid=\${api-key}&units=imperial</code>
	This tutorial describes how to return the temperature in degrees Fahrenheit. If you want, you can return the temperature in degrees Celsius by specifying the following value: <code>?appid=\${api-key}&units=metric</code>
Access type	Ensure Public gateway is selected. This access type uses the public internet to send messages.

- d. Click **Test** to confirm that you configured the connection successfully. When testing completes, the following message is displayed.

Connection Weather was tested successfully.

The API key is not validated as part of this test. You must validate the API key separately before using them with this security policy.

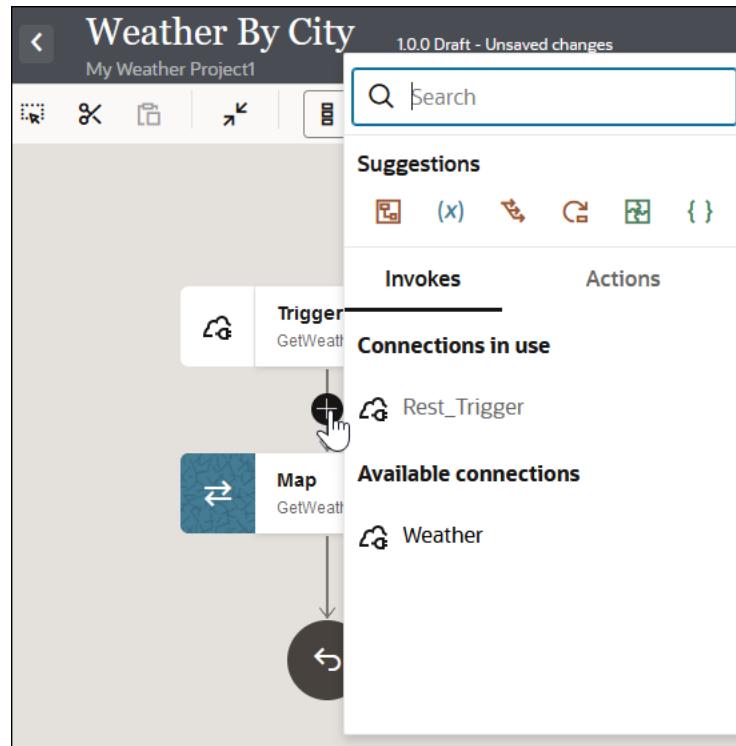
While the API key may be completely valid, it cannot be completely validated from within Oracle Integration. You have the option of completely validating the API key in [\(Optional\) Step 2. Test the API Key Outside of Oracle Integration](#).

- e. Click **Save**.
f. Click **Back** to return to the integration canvas.

Step 7. Specify How the Invoke Connection Retrieves the Latitude and Longitude Coordinates for the City

Let's now configure the REST Adapter invoke connection in the integration to retrieve the latitude and longitude coordinates for the city you specify when running the integration. During configuration, you specify details such as the operation, the relative resource URI, the query parameters, and the message payload content.

1. Drag your cursor along the line below the **GetWeatherByCity** trigger until the **Add** icon appears.
2. Select **Invokes**, then select the **Weather** connection.



The Adapter Endpoint Configuration Wizard opens.

3. Enter the following information.

Field	Description
What do you want to call your endpoint	Enter the following: GetCoord
What does this endpoint do	Enter the following: Get latitude and longitude for a city
What is the endpoint's relative resource URI	Enter the following: /geo/1.0/direct
What action do you want to perform on the endpoint	Select GET .
Select any options that you want to configure	Scroll to the bottom of the page and select the following options: <ul style="list-style-type: none"> • Add and review parameters for this endpoint • Configure this endpoint to receive the response

4. Leave all remaining fields as they are.

The completed Basic Info page looks as follows.

Basic Info

What do you want to call your endpoint? * [?](#)

What does this endpoint do? [?](#)

What is the endpoint's relative resource URI? * [?](#)

What action do you want to perform on the endpoint? * [?](#)

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.
Select any options that you want to configure:

- Add and review parameters for this endpoint
- Configure a request payload for this endpoint
- Configure this endpoint to receive the response

5. Click **Continue** to access the Request Parameters page.
 - a. In the **Specify Query Parameters** section, click **Add**.
 - b. Double-click the row in the **Name** column to enter a value.
 - c. Enter **q**, then select **string** from the **Data Type** list.

Specify Query Parameters		
		Add Remove
<input type="checkbox"/>	Name	Data Type
<input type="checkbox"/>	q	string

- d. Click outside the **q** row.
 - e. Click **Continue** to access the Response page.
6. Enter the following information.

Field	Description
Select the response payload format	Select JSON Sample.

Field	Description
<<< inline >>>	<p>Click this field and enter the following content, then click OK.</p> <pre>[{ "name" : "London", "local_names" : { "af" : "Londen", "ascii" : "London", "az" : "London", "ca" : "Londres", "da" : "London", "de" : "London", "en" : "London", "eu" : "Londres", "feature_name" : "London", "fi" : "Lontoo", "fr" : "Londres", "gl" : "Londres", "hr" : "London", "hu" : "London", "id" : "London", "it" : "Londra", "la" : "Londinium", "lt" : "Londonas", "nl" : "Londen", "no" : "London", "pl" : "Londyn", "pt" : "Londres", "ro" : "Londra", "sl" : "London", "tr" : "Londra", "zu" : "ILondon" }, "lat" : 51.5085, "lon" : -0.1257, "country" : "GB" }, { "name" : "London", "local_names" : { "ascii" : "London", "de" : "London", "en" : "London", "feature_name" : "London", "fi" : "London", "fr" : "London", "lt" : "Londonas", "nl" : "London", "pl" : "London", "pt" : "London" }, "lat" : 42.9834, "lon" : -81.233, "country" : "CA" }</pre>

Field	Description
	<pre>}, { "name" : "London", "local_names" : { "ascii" : "London", "en" : "London", "feature_name" : "London" }, "lat" : 39.8865, "lon" : -83.4483, "country" : "US", "state" : "OH" }, { "name" : "London", "local_names" : { "ascii" : "London", "en" : "London", "feature_name" : "London" }, "lat" : 37.129, "lon" : -84.0833, "country" : "US", "state" : "KY" }, { "name" : "London", "local_names" : { "ascii" : "London", "ca" : "Londres", "en" : "London", "feature_name" : "London" }, "lat" : 36.4761, "lon" : -119.4432, "country" : "US", "state" : "CA" }]</pre>

Element	Ensure response-wrapper is selected.
What is the media-type of Response Body	Ensure JSON is selected.

The completed Response page looks as follows.

Select the response payload format

JSON Sample

Sample Location [?](#)

Drag and Drop
Select a file or drop one here.

--OR-- enter sample JSON
[`<<< inline >>>`](#)

Element

response-wrapper

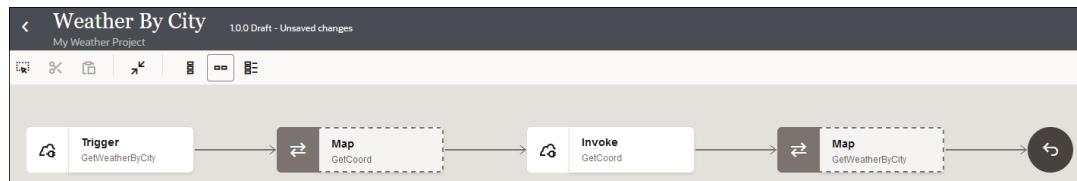
What is the media-type of Response Body? (Accept Header)

XML

XML(text)

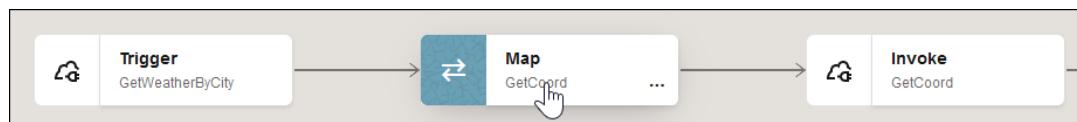
JSON

7. Click **Continue** to access the Summary page.
8. Review your selections, then click **Finish**.
9. Above the integration, click **Horizontal**  to see a horizontal view of the integration. Note that after you created and configured the **GetCoord** invoke connection in the integration, a map was automatically added after it.



When you configured the **GetWeatherByCity** trigger connection in the integration canvas, a map was automatically created. Let's now configure the first map.

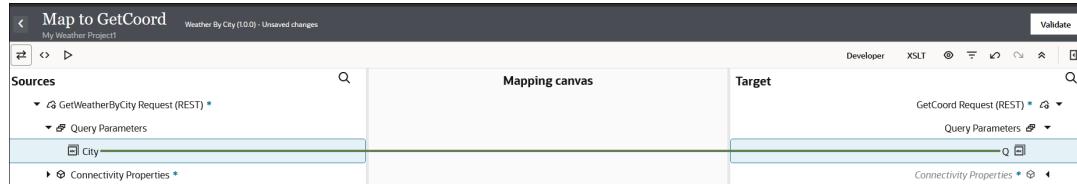
1. Double-click the map between the **GetWeatherByCity** trigger connection and **GetCoord** invoke connection.



The mapper opens.

2. In the **Sources** section, expand **Query Parameters**.
3. In the **Target** section, expand **Query Parameters**.

- Drag the source **City** element to the target **Q** element. This maps the city to its latitude and longitude coordinates. You created these query parameters earlier when you configured the trigger and invoke connections in the Adapter Endpoint Configuration Wizard.



- Click **Validate** and wait for the following message to appear.

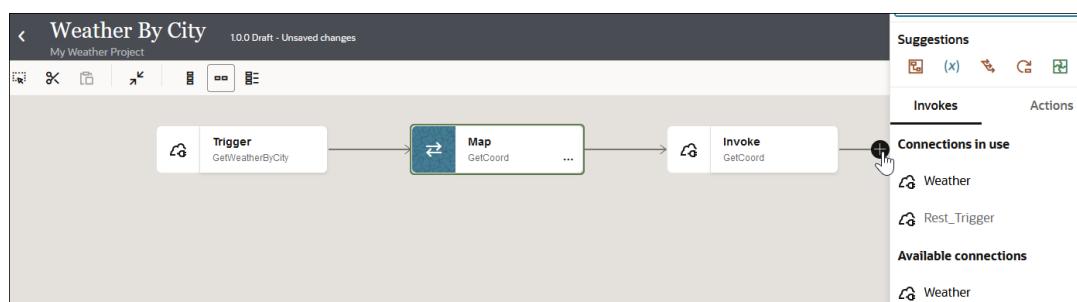
Map to GetCoord successfully validated.

- Click **Back** to exit the mapper and return to the integration canvas. Your mappings are automatically saved.

Step 8. Specify How the Invoke Connection Retrieves the Weather Report for the City

Let's now configure a second REST Adapter invoke connection in the integration to obtain the weather report from the OpenWeather application.

- Drag your cursor along the line after the **GetCoord** invoke connection until the **Add** icon appears.



- Select **Invokes**, then select the **Weather** connection again. The Adapter Endpoint Configuration Wizard opens.
- Enter the following information.

Field	Description
What do you want to call your endpoint	Enter the following: GetWeather
What does this endpoint do	Enter the following: Get the weather

Field	Description
What is the endpoint's relative resource URI?	Enter the following: /data/2.5/weather
What action do you want to perform on the endpoint?	Select GET .
Select any options that you want to configure	Scroll to the bottom of the page and select the following options: <ul style="list-style-type: none"> • Add and review parameters for this endpoint • Configure this endpoint to receive the response

4. Leave all remaining fields as they are.
 The completed Basic Info page looks as follows.

Basic Info

What do you want to call your endpoint? * [?](#)
GetWeather

What does this endpoint do? [?](#)
Get the weather

What is the endpoint's relative resource URI? * [?](#)
/data/2.5/weather

What action do you want to perform on the endpoint? * [?](#)
GET

Based on your selections, you can add parameters or configure a request and/or response for this endpoint.

Select any options that you want to configure:

Add and review parameters for this endpoint

Configure a request payload for this endpoint

Configure this endpoint to receive the response

5. Click **Continue** to access the Request Parameters page.
- In the **Specify Query Parameters** section, click **Add**.
 - Double-click the row in the **Name** column to enter a value.
 - Enter `lat`, then select **float** from the **Data Type** list.
 - Click **Add** again.
 - Double-click the row in the **Name** column to enter a value.
 - Enter `lon`, then select **float** from the **Data Type** list.

Specify Query Parameters

	Name	Data Type	Add	Remove
<input type="checkbox"/>	lat	float		
<input checked="" type="checkbox"/>	lon	float		

- g. Click outside the **lon** row.
 - h. Click **Continue** to access the Response page.
6. Enter the following information.

Field	Description
Select the response payload format	Select JSON Sample .

Field	Description
<<< inline >>>	<p>Click this field and enter the following content, then click OK.</p> <pre>{ "coord" : { "lon" : -122.08, "lat" : 37.39 }, "weather" : [{ "id" : 800, "main" : "Clear", "description" : "clear sky", "icon" : "01d" }], "base" : "stations", "main" : { "temp" : 282.55, "feels_like" : 281.86, "temp_min" : 280.37, "temp_max" : 284.26, "pressure" : 1023, "humidity" : 100 }, "visibility" : 10000, "wind" : { "speed" : 1.5, "deg" : 350 }, "clouds" : { "all" : 1 }, "dt" : 1560350645, "sys" : { "type" : 1, "id" : 5122, "message" : 0.0139, "country" : "US", "sunrise" : 1560343627, "sunset" : 1560396563 }, "timezone" : -25200, "id" : 420006353, "name" : "Mountain View", "cod" : 200 }</pre>

Element	Ensure response-wrapper is selected.
What is the media-type of Response Body	Ensure JSON is selected.

The completed Response page looks as follows.

Select the response payload format

JSON Sample

Sample Location [?](#)

Drag and Drop
Select a file or drop one here.

--OR-- enter sample JSON
[`<<< inline >>>`](#)
Element

response-wrapper

What is the media-type of Response Body? (Accept Header)

XML

XML(text)

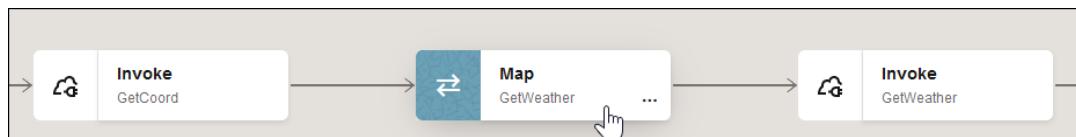
JSON

7. Click **Continue** to access the Summary page.

8. Review your selections, then click **Finish**.

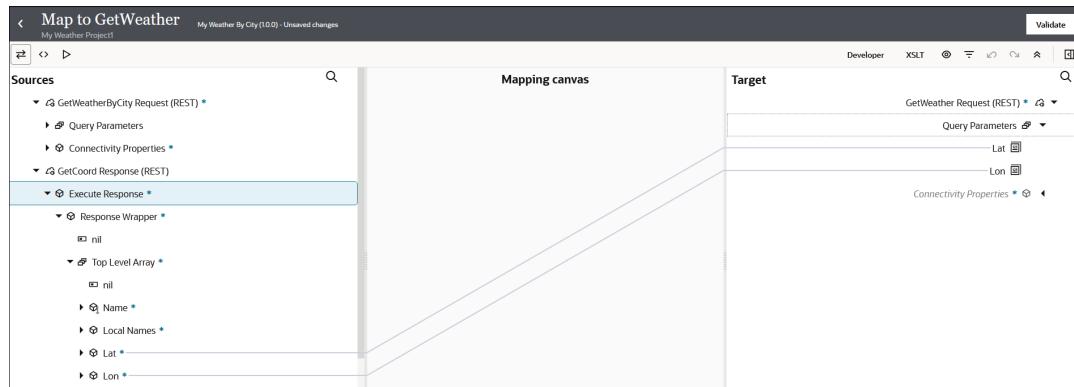
When you configured the **GetCoord** invoke connection in the integration canvas, a map action was automatically created. Let's now map the latitude and the longitude coordinates to the call to get the weather from the OpenWeather application.

1. Double-click the map between the **GetCoord** invoke connection and **GetWeather** invoke connection.



The mapper opens.

- In the **Sources** section, expand **GetCoord Response (REST)**, then **Execute Response**, then **Response Wrapper**, and then **Top Level Array**.
- In the **Target** section, expand **GetWeather Request (REST)**, and then **Query Parameters**.
- Perform the following mapping:
 - Drag the source **Lat** element to the target **Lat** element.
 - Drag the source **Lon** element to the target **Lon** element.



- Click **Validate** and wait for the following message to appear.

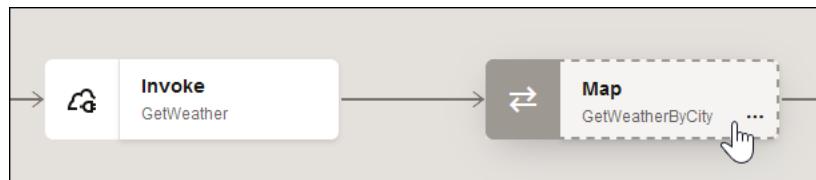
Map to GetWeather successfully validated.

- Click **Back** to exit the mapper and return to the integration canvas. Your mappings are automatically saved.

Step 9. Map the Weather Report Response Returned from the OpenWeather Application

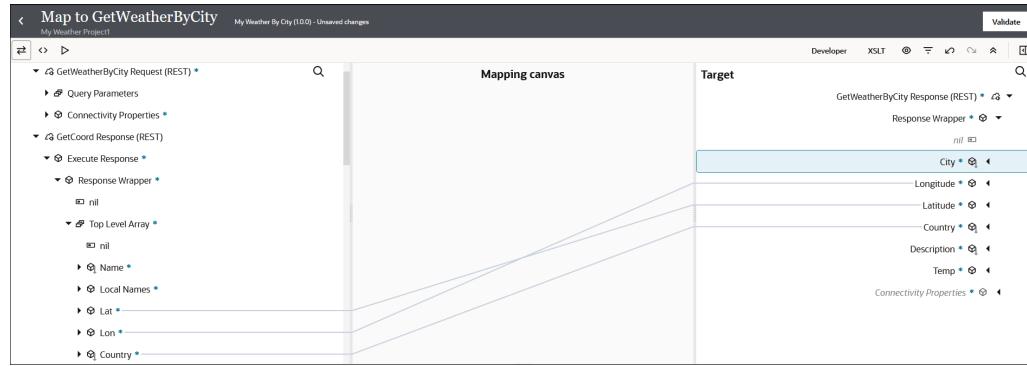
When you configured the **GetWeather** invoke connection in the integration canvas, a map action was automatically created. Let's now map the response information you receive back from the OpenWeather application.

- Double-click the map after the **GetWeather** invoke connection.

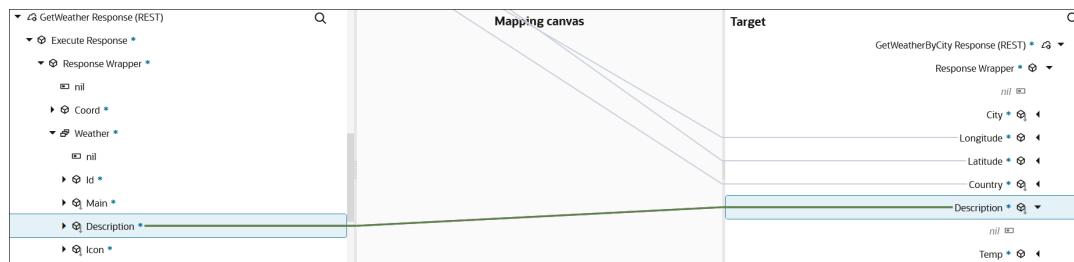


The mapper opens.

- In the **Sources** section, expand **GetCoord Response (REST)**, then **Execute Response**, then **Response Wrapper**, and then **Top Level Array**.
- In the **Target** section, expand **GetWeatherByCity Response (REST)**, and then **Response Wrapper**.
- Perform the following mapping:
 - Drag the source **Lat** element to the target **Latitude** element.
 - Drag the source **Lon** element to the target **Longitude** element.
 - Drag the source **Country** element to the target **Country** element.



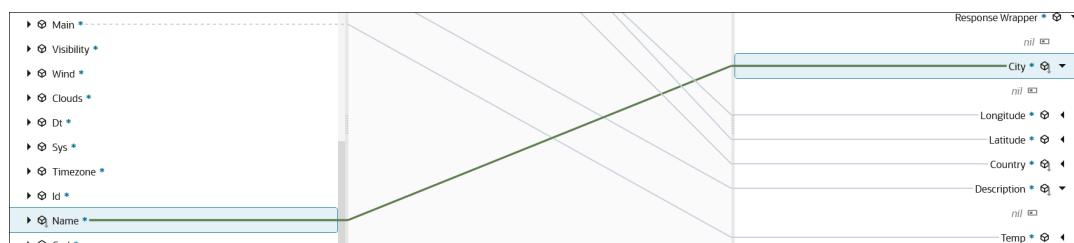
5. Scroll down in the **Sources** section and expand **GetWeather Response (REST)**, then **Execute Response**, then **Response Wrapper**, and then **Weather**.
6. Drag the source **Description** element to the target **Description** element.



7. Collapse the source **Weather** element in the **Sources** section.
8. Scroll down in the **Sources** section and expand the **Main** element.
9. Drag the source **Temp** element to the target **Temp** element.



10. Collapse the source **Main** element in the **Sources** section.
11. Scroll down in the **Sources** section and drag the source **Name** element to the target **City** element.



- Click **Validate** and wait for the following message.

Map to GetWeatherByCity successfully validated.

- Click **Back**  to exit the mapper and return to the integration canvas. Your mappings are automatically saved.

Step 10. Define a Business Identifier

Let's now define the primary business identifier for the integration. Business identifiers enable you to track payload fields in messages during runtime. Each integration requires a primary business identifier.

- Above the integration, click **Business Identifier** .

- Expand **QueryParameters**, and drag **city** to the **Business identifier** field.

- Click **Save**, then click outside the Business identifiers panel to close it. The business identifier you specify here is visible when you monitor your completed integration instance in the **Observe** tab in [Step 12. Monitor the Integration](#).
 - Check for any red error icons. If you see any, click the icon for details and resolve those errors.
- The complete integration design looks as follows.



- Click **Back**  to exit the integration canvas and return to the project page.

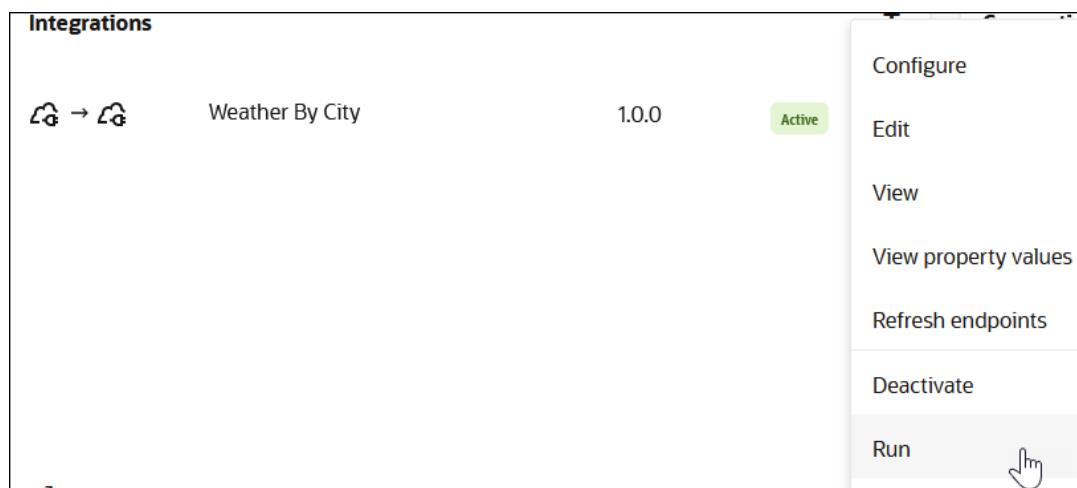
Step 11. Activate and Run the Integration

You are now ready to activate the integration to the runtime environment. Once activated, you can run the integration.

- In the **Integrations** section, click **Actions ...**, then select **Activate**.



2. Click **Activate**. Keep the tracing level set to **Production**. If you receive runtime errors, the **Debug (Not recommended)** selection provides detailed information in the activity stream and is useful for testing and troubleshooting any runtime errors.
3. Click **Refresh** periodically.
The integration status changes to **Active**.
You are now ready to run the integration and send a message.
4. In the **Integrations** section, click **Actions ...**, then select **Run**.



The Configure and run page enables you to run and test REST Adapter trigger-based integrations. This page provides a way of testing the delivery of an inbound message to your integration.

5. In the **Request** section, specify a city, and click **Run** to send your inbound message to the integration.

Configure and run

Configure request properties, then run the integration and verify the response. [Click here to read more.](#)

GET
`/ic/api/integration/v2/flows/rest/project/MY_WEATHER_PROJECT1/MY_WEATHER_BY_CITY/1.0/weather?city=Chicago`

Request

URI parameters Headers Body cURL Integration properties

city

The Activity stream panel opens and shows the movement of the message through the actions that you designed (triggers, invokes, and maps). The unique instance ID that was created for this integration run is displayed at the top. All milestones in this activity stream are green, indicating a successful run of this integration instance.

Activity stream

Instance Id: RkYKdhxfEe6w4tPcMLq93Q [Copy](#)

Tracing level: **Production** [Change](#) [Download](#)

Thu Jul 06 05:43:22.153 PM PDT 2023, 88ms	Wire Message received by Trigger GetWeatherByCity
05:43:22.241 PM, 1ms	Message started processing
05:43:22.242 PM, 384ms	Data Mapping completed
> 05:43:22.626 PM	Invoke GetCoord
05:43:22.781 PM, 239ms	Data Mapping completed
> 05:43:23.020 PM	Invoke GetWeather
05:43:23.070 PM, 3ms	Data Mapping completed
05:43:23.073 PM, 5ms	Message sent to reply
05:43:23.078 PM	Reply to GetWeatherByCity completed

6. View the response message at the bottom of the Configure and run page. The temperature of the city you specified is displayed (for this example, Chicago), along with its longitude and latitude coordinates, country, and current weather description.

Response

Status: 200 OK Instance ID: Vthg9BusEe6w4tPcMLq93Q [Copy](#)

Body Headers

```

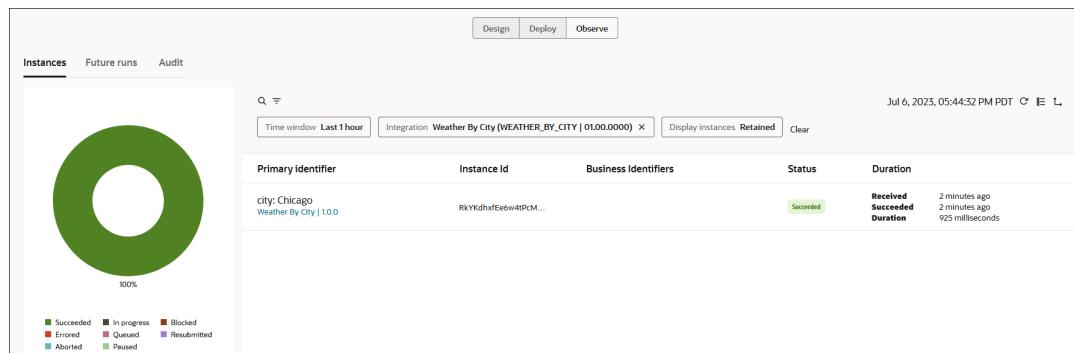
1 {
2   "city" : "Chicago",
3   "longitude" : -87.6244212,
4   "latitude" : 41.8755616,
5   "country" : "US",
6   "description" : "overcast clouds",
7   "temp" : 70.48
8 }
```

The response information you received back from the OpenWeather application is based on the target-to-source element mappings you performed in the **GetWeatherByCity** map in [Step 9. Map the Weather Report Response Returned from the OpenWeather Application.](#)

Step 12. Monitor the Integration

- Close the Activity steam panel and click **Track instances**.

The **Observe** tab in your project opens. This tab enables you to monitor the status of your integration instance. The status of the instance is displayed as **Succeeded**. Note that the primary business identifier (**city**) that you defined in [Step 10. Define a Business Identifier](#) is visible in the **Primary Identifier** column.



Congratulation! You have successfully created, designed, activated, run, and monitored your second integration. See the section below for more information about the capabilities used in this tutorial.

Learn More About The Features in this Tutorial

- [About Projects](#)
- [About Integrations](#)
- [About Connections](#)
- [Understand Trigger and Invoke Connections](#)
- [Map Data](#)
- [Assign Business Identifiers](#)
- [Activate an Integration](#)

- [Test Integrations from Outside the Integration Canvas](#)
- [Monitor Integrations in a Project](#)

Get Started with Projects

Oracle recommends designing your integrations in projects. Projects provide a single workspace for designing, managing, and monitoring integrations, robots, B2B, and healthcare components. Use projects to perform release management, manage access to automation solutions, and keep everything organized.

Topics:

- [About Projects](#)
- [Project Guidance If You've Used Packages](#)
- [Tour of the Projects Workspace](#)
- [10 Tips for Working with Projects](#)
- [4 Ways to Organize Projects](#)
- [About Project Deployment](#)
- [Restrictions for Projects](#)

About Projects

A project provides a single unified workspace for all stakeholders to design, manage, and monitor automation solutions, including integrations, robots, and healthcare components. Additionally, projects provide robust life-cycle management and streamlined updates to prebuilt integrations.

How Projects Help You

With everything at your fingertips, you'll build faster. And, with easy updates to integrations based on accelerators, you'll future-proof your prebuilt integrations. Finally, deploy quickly and with confidence using project deployments. Keep reading to learn more.

1. Manage the releases of your automation solutions

With project deployments, you can move integrations and robots to higher environments, such as testing or production, quickly and efficiently. A project deployment represents the current snapshot of all the integration and robot versions that must be in the environment you're promoting to. Enjoy improved stability in your higher environments and easier rollbacks when things go wrong.

See [About Project Deployment](#).

2. Control access to automation solutions and their components using fine-grained access control

With projects, you control the users and groups who can edit, view, and monitor an automation solution, including any integrations, robots, and healthcare components, using role-based access control (RBAC). Creating automation solutions within a project is the only way to control access at this fine-grained level.

See [Control Who Can Edit, View, and Monitor in a Project](#).

3. Build, manage, and monitor everything in one place

A project helps you get started quickly and confidently because you build all the components that an automation solution needs within a project. You don't need to click all over the user interface to find the right page to create a component. It's all right there within your project, including the ability to monitor your automation solutions.

How many integrations should you include in a project? What guardrails are in place to govern your use of projects? Get answers to these questions and more. See:

- [10 Tips for Working with Projects](#)
- [4 Ways to Organize Projects](#)
- [Restrictions for Projects](#)

4. Build faster by reusing components

After you've built an integration or two within a project, creating additional integrations is even faster and easier. The connections, JavaScript libraries, and lookups from existing integrations are all at your fingertips, offering easy one-stop shopping. The same is true for robots and healthcare components. Just grab what you need from the streamlined user interface and start building.

5. Future-proof your prebuilt integrations with easy updates

Accelerators provide prebuilt integrations that you can easily customize, but what happens to your customized integration when a new version of the accelerator is released? When you install the accelerator in a project, you can automatically update to the new version without reworking your customizations. Use accelerators with confidence, knowing that you can always accept the newest version without having to set aside time to modify and troubleshoot your integration.

See [Extend an Integration in an Accelerator Project](#) and [Upgrade an Accelerator Project and Merge Extensions](#).

6. Facilitate the exchange of business documents using projects

In a project, you can design, manage, and monitor integrations in B2B standalone mode. Use an AS2 Adapter connection as a trigger or an invoke connection, or use an FTP Adapter connection as an invoke connection. Track the processing of B2B messages as integration instances during runtime in the project.

See Introduction to B2B for Oracle Integration in *Using B2B for Oracle Integration 3*.

Ready to Convert Your Packages to Projects?

If you upgraded from Oracle Integration Generation 2, you can convert your packages to projects at any time. But first, familiarize yourself with the differences between packages and projects. See [Project Guidance If You've Used Packages](#).

Get Started with Projects

See [Design and Monitor Integrations in Projects](#).

Project Guidance If You've Used Packages

If you upgraded from Oracle Integration Generation 2, some of your integrations might be in packages. Learn the key differences between packages and projects and how to get started using projects.

Key Differences Between Packages and Projects

Area	Packages	Projects
Overall usage and goals	Packages primarily offer help with organization. For example, you can collect related integrations in a single package. You can also import and export packages, including their contents.	Projects offer many benefits, including improved release management, organization of artifacts, and a single unified workspace for designing, managing, and monitoring integrations. See About Projects .
Access control	Everyone who works in Oracle Integration can access the integrations in a package. Similarly, everyone can access integrations that aren't part of a package.	Projects offer fine-grained access control. You choose the people who can edit, view, and monitor the integrations and other assets in each project. See Control Who Can Edit, View, and Monitor in a Project .
Deployment	Creating a continuous integration and continuous deployment (CI/CD) pipeline and delivering updated integrations to higher environments requires work outside Oracle Integration. For instance, you can export a package and use the solution of your choice, such as Oracle Visual Studio, APIs, or shell scripts, to move the integrations into a testing or production environment.	If you've deployed integrations from packages before, deployment is easier and faster from a project. Projects provide built-in deployment capabilities for improved release management, including controlled deployment. Deploy your integrations where you design them, from within a project, without needing to create a pipeline or run a build. See Workflow for Deploying Automation Solutions .
Observability	Packages don't offer monitoring capabilities. You monitor integrations individually on the Observability pages.	You have two options for monitoring the integrations in a project: The Observe pages in the project and the Observability pages, which are outside the project. Monitoring within the project provides helpful context for the integrations.

Workflow for Converting a Package to a Project

If you used packages in Oracle Integration Generation 2, you can convert them to projects in just a couple of minutes using the package conversion tool.

The package conversion tool moves everything in the package into a project, including integrations, connections, lookups, and JavaScript libraries. The existing package remains untouched.

Note:

You can't convert accelerator and recipe packages into projects.

Step	Task	More information
1	Create a project from a package	Convert a Package to a Project
2	Control access to the project by selecting the people who can edit, view, and monitor the project	If you don't assign access, everyone's access within Oracle Integration determines the work they can do in the project. Control Who Can Edit, View, and Monitor in a Project

After you've created the project, you can deploy the integrations within it. See [Create and Manage a Project Deployment](#).

Workflow for Importing Standalone Integrations into Projects

Ready to start moving your standalone integrations into projects? All you need to do is spend a few minutes importing the integrations and their components into one or more projects.

Step	Task	More information
1	Create a strategy for organizing projects	4 Ways to Organize Projects
2	Create a project	Create or Import a Project
3	Export the integrations that you want to import into a project	Export an Integration
4	Add one or more integrations to the project	Create, Import, or Copy an Integration into a Project
5	Configure the properties for the connections in the integrations	Configure Connection Properties and Security Properties

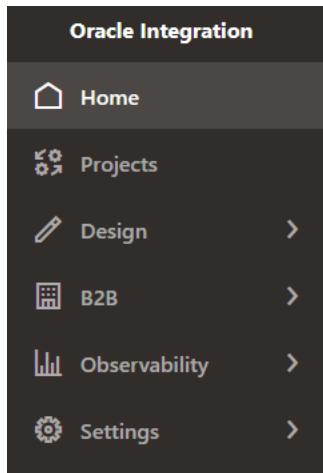
Tour of the Projects Workspace

The Projects workspace contains several tabs for designing, managing, and monitoring automation solutions.

Navigation to the Projects Workspace

A **Projects** option is available in the navigation pane.

In the Projects workspace, you can complete all your design, management, and monitoring tasks. You don't need to navigate to the **Design** and **Observability** pages when working in a project.



Tip:

When you move around the project workspace performing tasks, a **Back** button is provided that enables you to easily return to your previous location under the **Projects** option.

Design Tab

Design and manage integrations on the Design tab, which appears by default when you open a project. The design tab includes a vertical navigation bar to navigate among the pages.

The screenshot shows the Oracle Integration Design tab interface. At the top, there are tabs for Design, Deploy, and Observe. Below the tabs is a search bar with placeholder text "Full or partial name or keyword or description". The main area is divided into several sections:

- Integrations:** A list of integration components, each with a status (e.g., Configured) and three-dot menu. Examples include "Oracle Maintenance OFS Activity Type Lookup Helper", "Oracle OFS Maintenance Inventory List Helper", "Oracle Maintenance OFS Get Inventory Type Helper", and "Oracle OFS Maintenance Inventory Complete Helper".
- Connections:** A list of connection definitions, categorized by type (rest, erp). Examples include "rest Oracle Maintenance OAuth REST Connection", "rest Oracle Maintenance REST Connection", "rest Oracle Maintenance OFS REST Connection", and "erp Oracle Maintenance Adapter Connection".
- Lookups:** A section showing a single entry: "ORCL-BRT-ACT-TYPE-MAPPING" with status "Configured".
- Libraries:** A section showing a single entry: "inventorySync" with status "Locked".

The Design tab contains the following pages.

Integration Page

The following components are available on the Integration page on the Design tab.

Component	Description
Integrations section	<p>Create integrations in this section.</p> <p>Additional tasks that you perform in this section include:</p> <ul style="list-style-type: none"> • Viewing, editing, cloning, deleting, searching, activating, and deactivating integrations. • Enabling or disabling the ability for REST Adapter-triggered and AS2 Adapter-triggered integration instances to be replayable at runtime. • Creating a new version of an integration. • Setting tracing levels. • Refreshing endpoints. • Updating property values. • Running REST Adapter trigger-based integrations and schedule integrations. • Viewing run details.
Connections section	<p>Create connections for your integrations in this section.</p> <p>Additional tasks that you perform in this section include:</p> <ul style="list-style-type: none"> • Editing, deleting, and searching for connections. • Refreshing metadata.
Lookups section	<p>Create lookups for your integrations in this section.</p> <p>Additional tasks that you perform in this section include editing, deleting, and searching for lookups.</p>
Libraries section	<p>Create JavaScript libraries for your integrations in this section.</p> <p>Additional tasks that you perform in this section include deleting, updating, and searching for libraries.</p>
Events area	<p>Create events to publish and subscribe to in integrations.</p> <p>Additional tasks that you perform in this section include cloning, editing, deleting, and viewing events.</p>

Robot Page

The following components are available on the Robot  page on the Design tab.

For information about working with robots and their components, see Learn About Robots and Build a Robot in *Using Robots in Oracle Integration 3*.

Component	Description
Robots section	Create and edit robots, create new versions of robots, associate and remove environment pools, activate and deactivate robots, delete robots, and more in this section.
Environment pools section	Create and edit environment pools, add and remove environments from the pools, and more in this section.
Robot connection types section	Create, edit, and delete robot connection types in this section.
Robot connections section	Create, edit, and delete robot connections in this section.

Healthcare Page

The following components are available on the Healthcare  page on the Design tab.

Component	Description
HL7 messages and schemas	<p>Create HL7 messages and schemas in this section.</p> <p>See Create Oracle Integration for Healthcare Schemas and Documents in <i>Using Oracle Integration for Healthcare in Oracle Integration 3</i>.</p> <p>Additional healthcare-related tasks that you can perform include:</p> <ul style="list-style-type: none"> • Cloning, editing, viewing, and deleting schemas. • Cloning, editing, viewing, and deleting messages. • Exporting a project that includes healthcare resources (schemas, messages, and integrations using a healthcare action) and importing that project into another instance. • Cloning a project that includes healthcare resources (schemas, messages, and integrations using a healthcare action) <p>You can select the HL7 message when configuring the healthcare action in an integration in the Integrations section of the project. See Convert HL7 Messages with a Healthcare Action in <i>Using Oracle Integration for Healthcare in Oracle Integration 3</i>.</p>

B2B Page

The following components are available on the B2B  page on the Design tab.

Component	Description
Documents and schemas	<p>Create or import documents and schemas in this section.</p> <p>See B2B Schemas and B2B Documents in <i>Using B2B for Oracle Integration 3</i>.</p> <p>Additional B2B for Oracle Integration-related tasks that you can perform include:</p> <ul style="list-style-type: none"> • Editing, viewing, cloning, exporting, and deleting schemas, and generating an implementation guide. • Editing, viewing, cloning, exporting, and deleting documents. • Exporting a project that includes resources (schemas, documents, and integrations using a B2B action) and importing that project into another instance. • Cloning a project that includes B2B resources (schemas, documents, and integrations using a B2B action). <p>You can select the document when configuring the B2B action in an integration in the Integrations section of the project. See Use the B2B Action in Standalone Mode in <i>Using B2B for Oracle Integration 3</i>.</p>
Trading partners	<p>Create or import trading partners in this section.</p> <p>You can also edit, view, export, and delete trading partners.</p> <p>See Use B2B for Oracle Integration in Trading Partner Mode in <i>Using B2B for Oracle Integration 3</i>.</p>

Not all B2B features are currently supported in projects. See Design B2B Integrations in Projects or in Standalone Environments.

Components on All Pages

Some components are available on all pages in the Design tab.

Component	Description
View dependencies	View the dependent relationships between all integrations, connections, lookups, JavaScript libraries, and events in a project.
Edit details	View integration description details.
Export	Export a project.
Share	Assign a project owner and specify the people who can edit, view, and monitor resources in the project.
Refresh	Refresh the area you are in separately from other areas (for example, healthcare).

Deploy Tab

Deploy one or more automation solutions in a project on the Deploy tab.

The screenshot shows the Deploy tab interface. At the top, there are three tabs: Design, Deploy (which is selected), and Observe. Below the tabs, it says "Showing 1 deployment(s)". A table lists one deployment entry:

Name	Identifier	Flows	Last updated
1.0.1	0100.0001	1	Aug 2, 2024 08:07:05 AM PDT

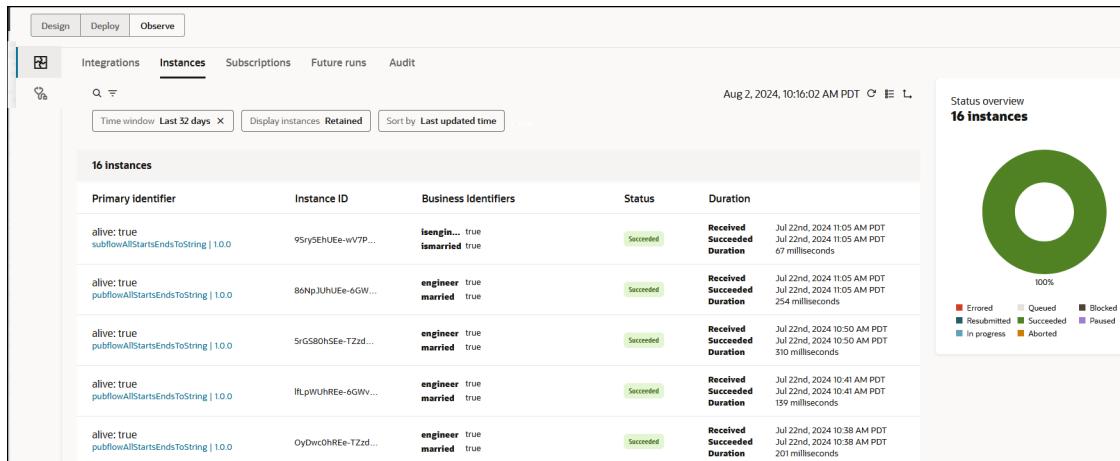
On the right side of the table, there is a "Create" button and a magnifying glass icon.

The Deploy tab contains the following components:

Component	Description
Create button	Create a project deployment.
Options that appear when you point to a project deployment	Contains options for exporting, activating, and deactivating a project deployment.
Actions *** menu , which appears when you point to a project deployment	Contains options for editing, viewing, cloning, and deleting a project deployment.

Observe Tab

Monitor your automation solutions on the Observe tab.



The Observe tab contains the following pages.

Integration Page

The following tabs are available on the Integration page on the Observe tab.

Tab	Description
Integrations tab	See high-level information about your integrations and their messages.
Instances tab	<p>This tab contains the following components:</p> <ul style="list-style-type: none"> A table for tracking the status of integration instances through business identifiers and viewing the activity stream and message payloads of an instance. <p>Use a clickable graph to filter the display of the state of instances.</p> <p>You can also use the Filter icon to perform the same task.</p> <ul style="list-style-type: none"> A section for viewing instance errors. You can resubmit failed instances, discard failed instances, view message recovery status, and view basic and detailed error messages. An Actions menu for each instance to perform other tasks such as viewing the integration schedule and child instances and replaying REST Adapter-triggered and AS2 Adapter-triggered integration instances that you configured as replayable during integration activation.
Subscriptions tab	Shows the currently subscribed integrations, the event, retained events, and subscriber status.
Future runs tab	View a calendar of scheduled integration runs.
Audit tab	View and download the design-time audit log of actions performed in a project.

Robot Page

The following tabs are available on the Robot page on the Observe tab.

Tab	Description
Robot instances tab	<p>This tab contains a table for tracking the status of robot instances through business identifiers and viewing the activity stream and message payloads of an instance.</p> <p>Use a clickable graph to filter the display of the state of instances.</p> <p>You can also use the Filter  icon to perform the same task.</p>

B2B Page

The following tabs are available on the B2B  page on the Observe tab.

Tab	Description
Business messages	<p>This tab contains the following components:</p> <ul style="list-style-type: none"> A filter for sorting on last message created, date range, trading partner name, message direction, message type, status, resubmit status, document status, document standard, message ID, wire message ID, transport message ID, integration instance ID, error code, document version, document type, business identifier, parent message ID, resubmitted message ID, collaboration ID, group control number, transaction control number, and interchange control number. A table for viewing the document, trading partner, document status, business identifier, and creation date. A View details  link for accessing business message details, such as properties, message tags, payloads, and correlations. An Actions  menu for viewing wire messages and functional acknowledgments.
Wire messages	<p>This tab contains the following components:</p> <ul style="list-style-type: none"> A filter for sorting on last message created, date range, trading partner name, message direction, message type, status, resubmit status, transport, message ID, transport message ID, integration instance ID, error code, payload name, parent message ID, and resubmitted message ID. A table for viewing the transport, trading partner, transport status, transport message ID, creation date, payload name, and payload size. A View details  for accessing wire message details, such as properties, message tags, payloads, and correlations. An Actions  menu for viewing business messages and the message disposition notification (MDN).

10 Tips for Working with Projects

Ready to be an expert at projects? These tips will help you get started on the right path.

1. Allow Your Business Goals to Inform Your Strategy for Projects

With a project, you can create a custom solution that supports your business goals. See [4 Ways to Organize Projects](#).

2. Aim for No More Than Twenty Integrations per Project

The optimal number of integrations for each project is the number that makes sense for your business goals. Oracle recommends limiting the integrations in a project to about twenty, at

least when you're starting out, so that the assets within a project are still easy to find and navigate at a glance. Whatever number you choose, make sure you stay within the service limits. See Service Limits in *Provisioning and Administering Oracle Integration 3*.

3. Remember That Organization Isn't Always About Separation

Managing related integrations in a project is useful, but what if you want to call a child integration from integrations that are in multiple projects? Not to worry, projects can accommodate this requirement, too. When you create an integration in a project, choose whether to make the integration available in other projects, in addition to its own project.

See [Invoke Child Integrations Inside or Outside of Projects](#) and [Create an Integration](#).

4. Keep Everything Secure by Giving the Minimum Access Required

The tenancy administrator assigns application roles that grant people access to work in Oracle Integration. The tenancy administrator assigns these roles in the Oracle Cloud Infrastructure Console.

Additionally, the project owner, who is typically either a developer or an administrator, manages access to projects using role-based access control (RBAC). The project owner assigns these roles in a project in Oracle Integration.

To work in a project, a person must have both the required application roles and the required project roles. Project roles never override application roles. As a best practice, administrators and project owners should give people only the minimum access they need to do their jobs.

Reminder: You can control access to projects, but the control extends only to design time, not to runtime. Those with the endpoint, the credentials, and an OAuth token can run the integration at runtime.

5. Plan Your Working Times if You're Working with a Team

Your entire team can work in a project at the same time, but the same is not true of integrations. When a person opens an integration for editing, the integration is locked, and no one else can edit it. Make sure you plan your work accordingly so that everyone can complete their work in a timely way.

6. Be Descriptive When Versioning and Deploying

When you version an integration and when you create a project deployment, make sure you include information about the changes in the Description field. This information helps you select the right integration version when you create the project deployment and select the right project deployment to export.

7. Adopt a Versioning Strategy for Integrations

Version numbers for integrations use the following format: xx.yy.zzzz, where each x, y, and z is a number that you specify.

Typically in versioning, the xx numbers correspond to major versions, the yy numbers correspond to minor versions, and the zzzz numbers correspond to patch versions. You can increment version numbers however you want, but Oracle recommends establishing standard operating procedures for when you increment each number. For instance:

- Increment the xx number by one when you add an integration to the project deployment. The endpoint URLs for the integrations in the project deployment don't change when you increment the version number.
- Increment the yy number by one for significant changes to integrations.

- Increment the zzzz number by one for minor changes.

8. Choose Carefully When Creating Project Deployments

Deploying multiple integrations at the same time is easy, thanks to fine-grained deployment creations in a project. All you need to do is select the integrations to include and their versions. Just make sure you select the correct integrations and versions to include in the project deployment.

See [About Project Deployment](#).

9. Update Your Workflows

Projects offer streamlined workflows for all your integration work, from designing to monitoring. Spend less time navigating from place to place in the user interface and more time meeting your organization's integration needs.

See [Workflows for Integrations](#).

10. Monitor Integrations on the Observe Page

When you want to see what's happening with all the integrations in a project, go to the Observe page within a project. Here, you can see at-a-glance status and error information for all of the project's integrations.

Additionally, as long as you have access to a project, you can monitor the integrations on the Observability pages.

4 Ways to Organize Projects

Ready to get started with projects, but want some organization inspiration? Keep reading for ways you can use projects to organize your integrations.

The number of projects that you create and the integrations that you include in each project is up to you, allowing you to create a custom solution that supports your business goals.

1. Organize by Product or Line of Business

When you organize by product, you can organize the work of each line of business that works in an instance. For example, create:

- One or more projects for ERP integrations
 - For example, one project for accounts receivable integrations in ERP, another for order capture in ERP, another for year end in ERP, and so on.
- One or more projects for HCM integrations
 - For example, one project for onboarding employees in HCM, and so on.
- Additional projects as needed.

2. Organize by Solution

When you organize by solution, all components related to a solution are in a single project.

For example, create one project for all components related to synchronizing Salesforce opportunities with Oracle Fusion Cloud Applications, and create additional projects for other solutions.

3. Keep Common Integrations Together

Projects optimize your day by keeping everything you need at your fingertips. If you have a number of logging assets or error handling assets, you can group them in one or two projects so that you can find them more easily.

For example, you might create one or more integrations that handle error handling and call the integrations from your other integrations. Keeping all the integrations related to error handling in a single project makes the integrations easy to find.

4. Mix and Match

Why limit yourself to one method of organization? Projects are highly customizable so that you can create as many as you need. Mix and match these options, or come up with your own ideas.

About Project Deployment

Use a project deployment to move automation solutions with integrations and robots to a higher environment, such as from development to testing. Project deployments offer convenience, easier activation of integrations, and improved stability of your testing and production environments.

If your process for deploying automation solutions to a new environment has focused on moving only updated integrations, a project deployment is going to offer new capabilities, including a more stable production environment. To take advantage of all the benefits of production deployments, you'll need to approach your promotion differently. Keep reading to learn more.

Think Beyond New and Updated Integrations and Robots

Before working in projects, when you needed to deploy a new or updated integration to a testing or production environment, you probably moved only the new or updated integration.

To take advantage of the convenience of project deployments, including easier activation of integrations, you can continue this practice and include only new and updated integrations in each project deployment. However, to gain improved stability in your testing and production environments, you should deploy differently when working with project deployments.

Project Deployments Are the Current Snapshot

Think of a project deployment as a snapshot of all the integrations that must be in the environment that you're promoting to. For instance, consider a project with fifteen integrations. You included ten of the integrations in an earlier deployment and promoted them to your testing environment. You're now ready to promote the remaining five integrations to the testing environment.

Before working in projects, you probably would have moved only the remaining five integrations to testing. However, the best practice for project deployments is to include the five integrations as well as the ten other integrations in the project deployment.

Enjoy Improved Stability and Easier Rollbacks

If the ten integrations in testing haven't changed, why should you include them in the project deployment? The reason is stability.

If you include all twelve integrations in the production deployment and experience any issues in the testing environment, you can immediately roll back to the previous stable version of your product deployment.

Issues in a testing environment are expected and likely aren't going to derail your business. However, imagine the power of a project deployment in your production environment. If a new or updated integration introduces any issues, you can immediately roll back to the previous version of the project deployment. Your business will stay on track, and you can troubleshoot and resolve any issues in your lower environments.

Workflow for Deploying Automation Solutions

When your automation solutions are finished, a few fast and easy steps promote the integrations, robots, and more to your testing or production environments. See [Workflow for Deploying Automation Solutions](#).

Restrictions for Projects

Be aware of the restrictions for projects.

Contents of a project

- There are service limits on the number of integrations, project deployments, connections, lookups, and JavaScript libraries you can create in a project. See Service Limits in *Provisioning and Administering Oracle Integration 3*.

To stay within the project deployment limit, delete old versions periodically. If you need to keep older versions for auditing purposes, you can export the project deployments and archive them using your organization's archiving tools, such as a source control system.

- Lookups and JavaScript libraries are visible and usable only in a single project.

Your uncluttered view shows you only the assets in which you are interested.

Similarly, assets created in standalone integrations or integrations in packages are not visible or usable in a project.

- You can use certificates, which are defined globally, from any project.

Security and other restrictions

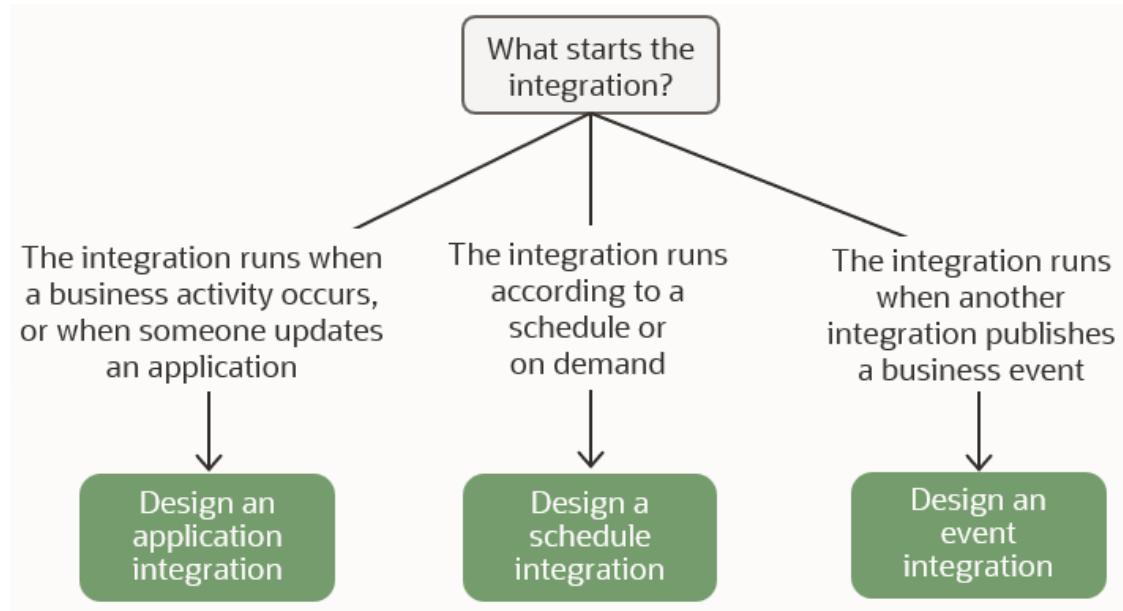
- You cannot invoke one project from another project.
- You cannot migrate a package of integrations to a project.
- You cannot clone individual JavaScript libraries within a project.

Understand Integration Patterns

You can select from several patterns when creating an integration in the Create integration panel.

What Kind of Integration Should You Design?

When you create an integration, the first decision to make is the integration model to use. The main difference among the models is the way that they start running.



Available Integration Patterns

Pattern	Description
Application	<p>Create an application integration (previously known as app-driven orchestration) that uses an event or a business object to trigger the integration. For example, you create an integration with an Oracle Service Cloud (RightNow) Adapter as a trigger and an Oracle CX Sales and B2B Service Adapter as an invoke. The Oracle Service Cloud (RightNow) Adapter subscribes to an event from the Oracle CX Sales and B2B Service application to trigger the integration. Application integrations include many actions for performing specific tasks, such as the following:</p> <ul style="list-style-type: none"> • Switch actions to create multiple routing expressions. • For-each actions for looping over repeating elements. • Assign actions for assigning values to scalar variables. • Ad-hoc mappings on switch actions branches. • Callback actions (to end a process and respond back to the sender) and end actions (to end a process without responding back to the sender) in asynchronous integrations. <p>See About Application Integrations and Create Application Integrations.</p>
Schedule	<p>Create an integration that uses a schedule to trigger the integration instead of an adapter. For example, you add an initial invoke adapter to read a trigger file and a second FTP adapter to download the file for further processing. After designing this integration, you schedule when to run it.</p> <p>See About Schedule Integrations and Create Schedule Integrations.</p>
Event	<p>Create and select events for publishing and subscribing to in integrations. You define the events in JSON-formatted or XML schema files.</p> <p>Note: If you created integrations using the Publish to OIC and Subscribe to OIC integration patterns (often called pub/sub) in Oracle Integration Generation 2, these capabilities and more are available in Oracle Integration 3. To create an integration that publishes an event, create an application integration. To create an integration that subscribes to published events, create an event integration. See About Event Integrations Create Event Integrations.</p>

About Application Integrations

An application integration runs when a business activity occurs, such as a customer submitting an order; or when someone adds or changes something in an integration. Application integrations can be synchronous or asynchronous, depending on whether they provide an immediate response.

Quick Introduction to Sync and Async

- **If you require real-time information, create a synchronous integration.**
For the sake of efficiency, you might want to pair it with an asynchronous integration.
- **To perform work in batch operations, create an asynchronous integration.**
An asynchronous integration is sometimes called fire and forget.

Real-World Analogy: Phone versus Email

Consider a situation that you've probably encountered: You need a coworker who is in a different location to complete a task for you.

You have several options for contacting your coworker:

- **If the task is urgent and time sensitive, you call your coworker**
You need to know that they received your request and that they'll start and finish the task quickly.
In the integration world, this scenario is **equivalent to a synchronous integration**. Synchronous integrations send immediate responses and run quickly, usually in seconds or less.
- **If the task isn't urgent or time sensitive, you email your coworker**
You won't get a confirmation that they have read the message, but you know that your coworker is very responsive to requests. They'll fit the task into their schedule as soon as they can.
In the integration world, this scenario is **equivalent to an asynchronous integration**. Oracle Integration adds the requests for asynchronous integrations to a queue and processes the requests as soon as it can.

Learn More About Synchronous and Asynchronous Integrations

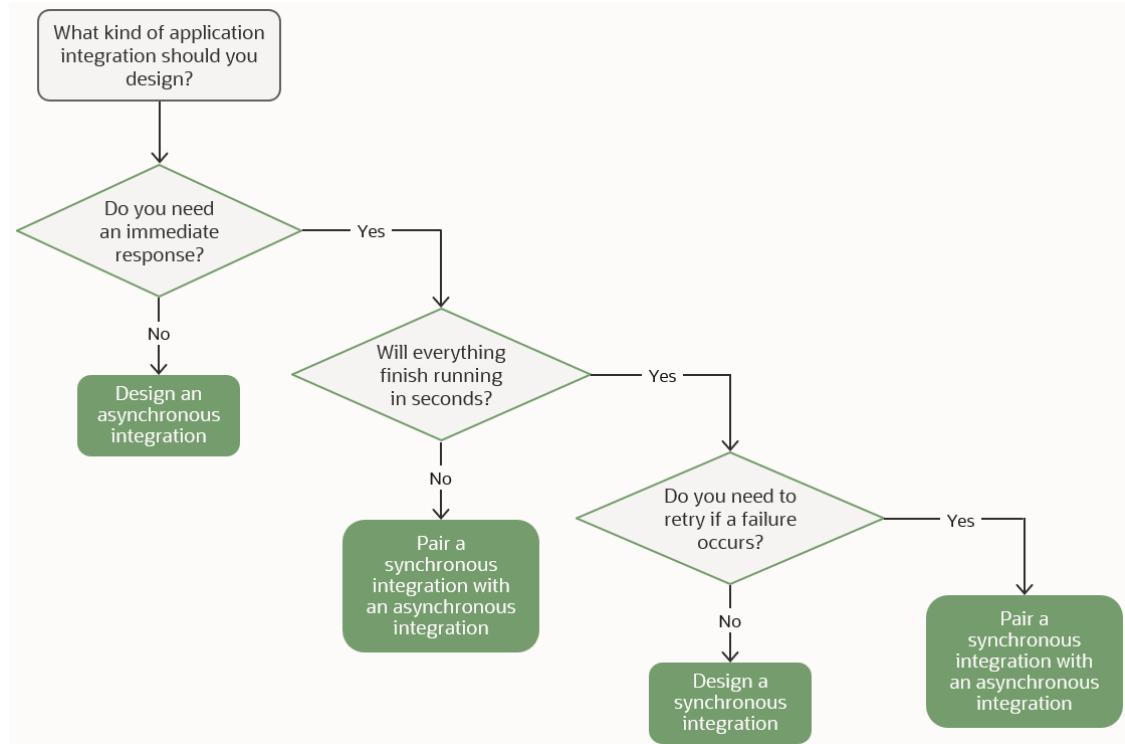
- [How to Choose Between Synchronous and Asynchronous Integrations](#)
- [How to Specify if an Integration is Synchronous or Asynchronous](#)
- [Differences Between Synchronous and Asynchronous Integrations](#)

How to Choose Between Synchronous and Asynchronous Integrations

An application integration can be synchronous or asynchronous. Learn when to create an asynchronous integration, a synchronous integration, or a combination of both.

What Kind of Application Integration Should You Design?

You have several options for application integrations: They can be synchronous, asynchronous, or a combination. When you know you need an application integration, the following decision tree helps you choose the type of application integration.



Detailed Explanation

The following table walks you through the questions in the previous decision tree and provides additional context for the questions and answers.

Question and details	Examples	Next steps
Do you need an immediate response? Sometimes, the client application that starts the integration requires an immediate response.	No, I don't need an immediate response If you're updating a record in an application, you typically don't need an immediate response. For example: <ul style="list-style-type: none"> • You need to synchronize inventory between two applications. • You need to create a packing list for an order. • You need to create a new employee's accounts in various business systems after you add a new record for them in your human capital management (HCM) software. Yes, I need an immediate response <ul style="list-style-type: none"> • A customer places an order on your website and wants to see an order confirmation page right away. • You must complete a credit check for a customer before fulfilling an order. 	No, I don't need an immediate response Create an asynchronous integration. You can stop reading here. Yes, I need an immediate response Go to the next question.
Will everything finish running in seconds? Synchronous integrations should finish in milliseconds or seconds. Asynchronous integrations are better suited to more complex or longer-running integrations, such as integrations that must process hundreds of records. To make this decision, draw upon your and your coworkers' experience with building integrations. If everyone is new to building integrations, consider building the integration, testing it under real-world circumstances, and checking the integration's performance in the activity stream. If the integration's running time is too long, you can always move some of the logic into a separate synchronous integration.	No, the integration can't run in seconds Consider an online retailer that collects all orders in a single file. One time per day, they run an integration that loops through the orders and processes them. The processing work includes validation, enrichment, and posting to an order-processing application. This integration takes several minutes to run. Yes, the integration can run in seconds Consider a bank that lets customers check their bank balances in a mobile app. The customers want to know their balances immediately, not in a minute or two.	No, the integration can't run in seconds Create a synchronous integration that provides an immediate response, and use it to run an asynchronous integration that performs the longer or more complex task. For example, when a synchronous integration provides a customer with an order confirmation number, the synchronous integration can kick off an asynchronous integration that completes the remaining order-related tasks, including creating an account for the customer, registering the customer for marketing emails, updating inventory, and generating a packing list, and sending the packing list to the warehouse. Yes, the integration can run in seconds Go to the next question.

Question and details	Examples	Next steps
<p>Do you need to retry if a failure occurs?</p> <p>Integration failures typically occur when the client application that the integration connects to goes down. Synchronous and asynchronous integrations handle failures differently:</p> <ul style="list-style-type: none"> When a synchronous integration fails, Oracle Integration returns a failure response to the client. You cannot rerun the integration. Sometimes, you don't want to rerun an integration, typically because the information that the integration obtains is time sensitive and is relevant only to an immediate need. Oracle Integration saves all requests for asynchronous integrations. Therefore, if the client application is down or otherwise unavailable when Oracle Integration sends a request for an asynchronous integration, you can rerun the integration instance in the future. 	<p>No, I don't need to rerun the integration</p> <p>Rerunning an integration doesn't make sense for the following situations:</p> <ul style="list-style-type: none"> Obtaining the current price for a stock ticker. Getting the current inventory numbers for an item. Determining how many seats are currently available for a show. <p>Yes, I need to rerun the integration</p> <p>Reunning an integration makes sense for the following situations:</p> <ul style="list-style-type: none"> Updating inventory in an application after the application went down for maintenance. After the application is available again, you need to rerun the integration so that your inventory can be synchronized for the start of business tomorrow. Attaching a document to a new employee's record after someone creates the employee's record in the HCM software. <p>After the employee record is created, you need to be able to attach the document.</p>	<p>No, I don't need to rerun the integration</p> <p>Create a synchronous integration.</p> <p>Yes, I need to rerun the integration</p> <p>Create a synchronous integration that provides an immediate response, and use it to run an asynchronous integration that contains the actions that you need to rerun.</p>

How to Specify if an Integration is Synchronous or Asynchronous

When you define the trigger for an integration, you must determine whether to configure the endpoint to receive a response. This selection determines whether the integration is synchronous or asynchronous.

- When you configure the endpoint to receive a response, the integration becomes synchronous.
- When you don't configure the endpoint to receive a response, the integration becomes asynchronous.

Differences Between Synchronous and Asynchronous Integrations

An application integration can be synchronous or asynchronous. Synchronous and asynchronous integrations have some key differences, including whether they provide an immediate response.

Question	Synchronous integration	Asynchronous integration
Optimal use	Best for delivering real-time information	Best for performing work in batch operations
Response	Oracle Integration sends a response after the integration finishes running, so you know the integration succeeded	Oracle Integration sends a response immediately after receiving the request, so you know Oracle Integration received the request The response is an HTTP 202 Accepted message You don't receive a response when the integration finishes running unless you configure an invoke to provide a notification
Run timing	The integration runs immediately	Oracle Integration places the integration in a queue
Payload	 The response includes a payload	 The response doesn't include a payload
Guaranteed delivery of requests	 The integration doesn't guarantee the delivery of its requests	 The integration guarantees delivery of its requests
Long-running tasks	 The integration cannot perform long-running tasks	 The integration can perform long-running tasks
Subscribe to events	 The integration cannot subscribe to events	 The integration can subscribe to events
Rerun after failure	 You cannot rerun the integration if it fails	 You can rerun the integration if it fails

About Schedule Integrations

A schedule integration starts moving data when you tell it to, either on the day and time of your choice or on demand. You can specify a regular cadence for the integration, such as a daily, twice daily, weekly, or any schedule that your business requires.

When to Use Schedule Integrations

A schedule integration solves a number of business problems. For example:

- Every day at 5 PM, synchronize your inventory in all of your applications.

- Four business days before every pay date, run payroll.

When Not to Use Schedule Integrations

Don't use a schedule integration to get real-time data. If you need real-time data, create an application integration that is synchronous.

About Event Integrations

An event integration subscribes to events that you create and publish in a separate integration. You create the events you want to publish and subscribe to in JSON-formatted or XML schema files.

Advantages of Event Integrations

- Integrations are decoupled to make them easier to maintain. Event messages move between different integrations without the integrations being aware of each other.
- Integrations are easily scalable. You can add additional subscribers to an existing event without concern for interrupting the existing event flow between publishers and subscribers.
- Events are made available to multiple integrations for consumption, allowing applications across your organization to simultaneously receive relevant payload.
- Custom headers can be defined in events, meaning that if the filter condition defined for the header is satisfied at runtime, the subscribing integration is triggered. If the filter condition is not satisfied, the integration is not triggered.

Event Design Process

You follow a simple design process for events in Oracle Integration:

- Create an event in JSON or XML schema format. You can optionally define custom headers in events, meaning that if the filter condition defined is satisfied at runtime, the subscribing integration is triggered.
- Create a publishing integration (application or schedule) in which you add a **Publish event** action for selecting the event.
- Create an event integration to subscribe to the event.

Learn More About Event Integrations

See [Create Event Integrations](#).

Understand Hybrid Integrations

You can configure access to resources residing in an on-premises network or private resources that are available in your virtual cloud network (VCN) in Oracle Integration.

Topics:

- [About Connecting to On-Premises Applications](#)
- [About Connecting to Private Resources](#)

About Connecting to On-Premises Applications

Use the connectivity agent to configure access to resources residing in an on-premises network in Oracle Integration.

Topics:

- [About Creating Hybrid Integrations](#)
- [About the Connectivity Agent](#)
- [Connection Patterns for Hybrid Integrations](#)

About Creating Hybrid Integrations

Many business use cases require integration between applications hosted on *public cloud* and resources residing in an *on-premises network* or *private cloud*. To facilitate such hybrid integrations, Oracle Integration provides the necessary infrastructure and architecture patterns.

For example, consider a business case where a quote or sales order configured through an Oracle Configure Price Quote (CPQ) application must be sent to an Oracle E-Business Suite application, hosted in an on-premises network, for creation and fulfillment of the sales order.

Capabilities of Hybrid Integrations

This type of hybrid integration enables you to have flows hosted on Oracle Integration that:

- Access SOAP/REST endpoints exposed by applications such as Oracle E-Business Suite, Siebel, and JD Edwards, and any on-premises home grown SOAP/REST APIs
- Access non-HTTP-based endpoints such as databases, JMS, AQ, local file systems, SAP, and others

Use Cases

These capabilities enable you to implement use cases such as the following:

- Send requests from a cloud application (for example, send a create service order request from an Oracle Service Cloud application) to an on-premises E-Business Suite application
- Synchronize bulk data extracts of a product from a product data hub in Oracle ERP Cloud with an on-premises Oracle database or an Oracle Database Cloud Service instance
- Synchronize customers that are added/updated in an on-premises SAP application with SaaS applications such as Oracle CX Sales and B2B Service Adapter, Oracle CPQ, Oracle Service Cloud, and Salesforce.com

Use the Connectivity Agent in Hybrid Integrations

Oracle Integration provides a component called the *connectivity agent* to facilitate hybrid integrations. See [About the Connectivity Agent](#).

Connection Patterns

For different connection patterns you can use to create hybrid integrations, see [Connection Patterns for Hybrid Integrations](#).

About the Connectivity Agent

Using the connectivity agent, you can create hybrid integrations and exchange messages between applications in private or on-premises networks and Oracle Integration. The connectivity agent framework extends reach of Oracle Integration to non-internet accessible systems. The connectivity agent also provides multithreading support, which allows for multiple executors to perform downstream message processing.

For supported connectivity agent message payload sizes, see Service Limits in *Provisioning and Administering Oracle Integration 3*.



Connectivity Agent Components

The connectivity agent consists of the following components.

Component	Description
Cloud agent	This agent is installed and runs in Oracle Integration and supports communication with on-premises applications.
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 to your on-premises environment for installation. A connectivity agent can support multiple target systems. An Oracle Integration instance can support multiple agent groups, each with up to two agents, in a cloud/on premises topology. The on-premises connectivity agent establishes connections with the cloud agent for message processing.

Connectivity Agent Functionality

The connectivity agent provides the following functionality.

Note:

While multiple connectivity agents can run on a single host, this is *not* the recommended practice. If you follow this practice, you must ensure that the physical host has enough resources to run multiple connectivity agents.

Feature	More information
Network	<ul style="list-style-type: none"> The connectivity agent opens no ports on the on-premises system for communication. The on-premises connectivity agent can use a proxy to access the internet (the same proxy as other internal applications and browsers use). Authentication support for outbound proxy access is provided.
Communications security	<ul style="list-style-type: none"> All communication is secured using TLS. The on-premises connectivity agent registers with Oracle Integration over TLS using OAuth with the connectivity agent-specific authentication generated by Oracle Integration. The on-premises connectivity agent processes requests by pulling messages from Oracle Integration across TLS. The on-premises connectivity agent posts responses by pushing messages to Oracle Integration across TLS.
Requests	<ul style="list-style-type: none"> The on-premises connectivity agent checks for work by making outbound requests through the firewall. All communication is initiated by the on-premises connectivity agent.
Connection details	The on-premises connectivity agent connections are configured by the agent retrieving the configuration details from Oracle Integration.
Adapter support	Adapters such as the Oracle E-Business Suite Adapter, Oracle JD Edwards EnterpriseOne Adapter, Oracle Utilities Adapter, and Oracle Siebel Adapter even when enabled with the connectivity agent invoke Oracle Integration for direct message processing when invoking integrations.
Data security	No data is persisted in the on-premises agent.
Additional details	<ul style="list-style-type: none"> The on-premises connectivity agent does not support trigger/polling for REST/SOAP endpoints. No existing J2EE container is required to deploy the on-premises connectivity agent.

Adapter Connections that Work with the Connectivity Agent

Any callbacks using REST/SOAP do not go through agent, but go directly to Oracle Integration. The on-premises connectivity agent works with the following adapter connections.

- Outbound (invoke) adapters**

The following adapters can be configured as invoke connections in an integration to support communication with endpoint applications:

- Amazon Simple Notification Service (SNS) Adapter
- Amazon Simple Queue Service (SQS) Adapter
- Amazon Simple Storage Service (S3) Adapter
- Apache Kafka Adapter
- Using the ArcGIS (ESRI) Adapter with Oracle Integration 3*
- Azure Active Directory Adapter
- Azure Event Grid Adapter
- Azure Service Bus Adapter
- Azure Storage Adapter
- Confluent Adapter
- File Adapter

- FTP Adapter
 - GraphQL Adapter
 - IBM DB2 Adapter
 - IBM MQ Series JMS Adapter
 - Microsoft SQL Server Adapter
 - MySQL Adapter
 - Netezza Adapter
 - OData Adapter
 - Oracle Advanced Queuing (AQ) Adapter
 - Oracle Autonomous Data Warehouse Adapter
 - Oracle Autonomous Transaction Processing Adapter
 - Oracle Database Adapter
 - Oracle Database Cloud Service Adapter
 - Oracle E-Business Suite Adapter
 - Oracle JD Edwards EnterpriseOne Adapter
 - Oracle Primavera P6 EPPM Adapter
 - Oracle Siebel Adapter
 - Oracle SOA Suite Adapter
 - Oracle Utilities Adapter
 - Oracle WebLogic JMS Adapter
 - PostgreSQL Adapter
 - REST Adapter
 - SAP Adapter
 - SAP ASE (Sybase) Adapter
 - Snowflake Adapter
 - SOAP Adapter
- **Inbound (trigger) adapters**

The following adapters can be configured as trigger connections in an integration. Trigger adapter use with the connectivity agent is accomplished with polling or direct message processing:

- Polling:
 - * Apache Kafka Adapter
 - * Azure Active Directory Adapter
 - * Azure Service Bus Adapter
 - * Confluent Adapter
 - * File Adapter
 - * IBM DB2 Adapter
 - * IBM MQ Series JMS Adapter

- * Microsoft SQL Server Adapter
- * MySQL Adapter
- * Netezza Adapter
- * Oracle Advanced Queuing (AQ) Adapter
- * Oracle Autonomous Data Warehouse Adapter
- * Oracle Autonomous Transaction Processing Adapter
- * Oracle Database Adapter
- * Oracle Database Cloud Service Adapter
- * Oracle WebLogic JMS Adapter
- * PostgreSQL Adapter
- * SAP Adapter
- * SAP ASE (Sybase) Adapter
- Direct message processing:
 - * Oracle E-Business Suite Adapter
 - * Oracle JD Edwards EnterpriseOne Adapter
 - * Oracle Siebel Adapter
 - * Oracle Utilities Adapter

Connection Patterns for Hybrid Integrations

Use the connectivity agent in any of the following patterns to set up a connection between an application on your private (or on-premises) network and Oracle Integration.

 **Note:**

You are responsible for configuring your own network routing rules. If you don't know how to configure these rules, contact your network team for assistance.

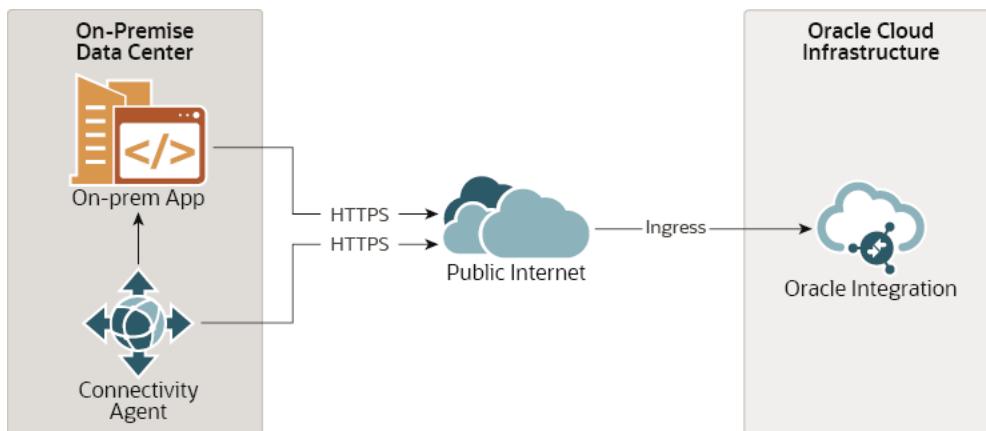
You can set up a connection over the public internet or select to configure an exclusive connection using FastConnect, which provides a faster, more reliable networking experience compared to the internet. You'll use the connectivity agent to communicate with Oracle Integration irrespective of the connection pattern you select; employing FastConnect only ensures that the traffic between your private (on-premises) network and Oracle Integration doesn't go over the public internet and remains private.

Public Internet Pattern

Install the connectivity agent on your private (on-premises) network. The inbound and outbound traffic to Oracle Integration goes over the public internet. For the outbound traffic from Oracle Integration, the connectivity agent initiates a secure connection to Oracle Integration, retrieves the request, and then invokes the required API in the on-premises application.

When you employ the connectivity agent, you don't have to open firewalls to access applications on your private network. In addition, all messages between the private network and Oracle Integration are encrypted.

To install and configure the connectivity agent, see [Manage the Agent Group and the On-Premises Connectivity Agent](#).



Oracle Cloud Infrastructure Only - Virtual Cloud Network Pattern

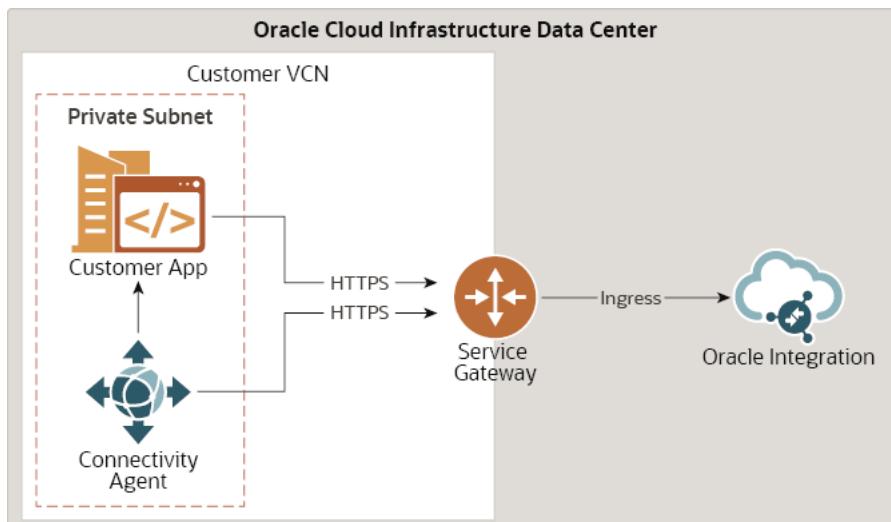
Install the connectivity agent in your Virtual Cloud Network (VCN) within Oracle Cloud Infrastructure, and configure a service gateway to route the traffic from the VCN to Oracle Integration.

Use this pattern if you have applications, such as Oracle E-Business Suite, running in a private subnet within Oracle Cloud Infrastructure. In this case, all traffic is routed locally and the public internet is not involved.

While not strictly required, it is highly recommended that all access from the VCN go through the service gateway. Service gateways primarily ensure that access to Oracle-hosted services is routed over the internal network. Users are not charged for service gateways. Service gateways only work within a region, and not across regions. For access across regions, traffic is still routed through a NAT gateway.

A service gateway is a common configuration for users that have implemented FastConnect or VPN/IPsec private peering to route traffic to Oracle Integration (ingress), including the connectivity agent in a private subnet.

For details on configuring a service gateway, see [FastConnect and VPN with Oracle Integration Cloud \(OIC\)](#).



FastConnect Public Peering Pattern

Install the connectivity agent on your private (on-premises) network, and set up an exclusive connection between your network and Oracle Integration using a FastConnect public peering link. The inbound and outbound traffic to Oracle Integration goes through the FastConnect link.

This connection pattern provides a faster and more reliable networking experience compared to the public internet pattern.

Follow the steps listed here to configure this pattern:

1. Subscribe to FastConnect with the public peering option. Currently, Oracle Integration directly supports only public peering with FastConnect. If you want to use the private peering option, you'll need to additionally use a VCN and a service gateway. See the [FastConnect Private Peering Patterns](#).

For detailed information on requirements and best practices for setting up an Oracle Cloud Infrastructure FastConnect, see FastConnect.

2. Configure your private (on-premises) network to route traffic through FastConnect.

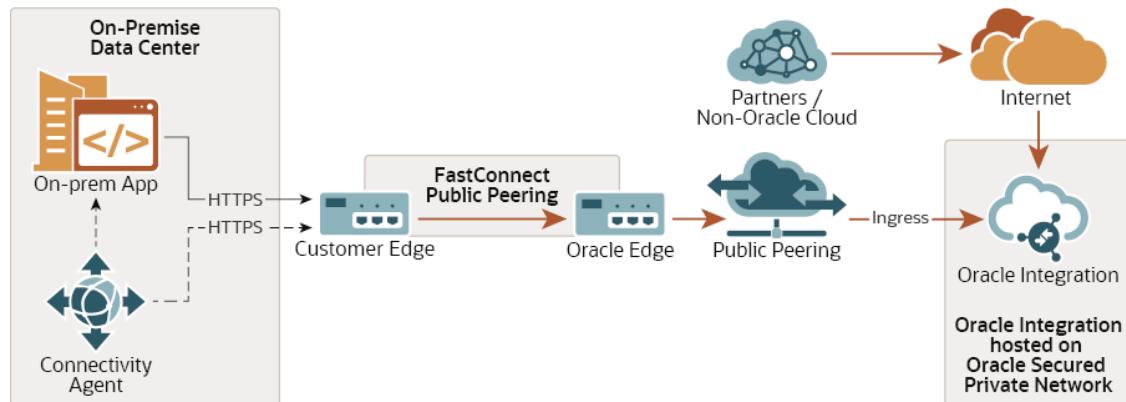
The FastConnect link contains the public IP addresses of Oracle Integration.

3. Finally, configure the connectivity agent to handle the outbound traffic from Oracle Integration to the on-premises application.

The connectivity agent also acts as a client to FastConnect and uses public peering.

 **Note:**

With FastConnect public peering, you must deploy the connectivity agent in your private or on-premises data center.



FastConnect Private Peering Patterns

In addition to providing fast and reliable connectivity, the FastConnect private peering patterns provide additional security to prevent traffic analysis.

Note:

The private peering patterns also apply to Virtual Private Networks (VPN) and are identical except that the FastConnect private peering link is replaced with a VPN.

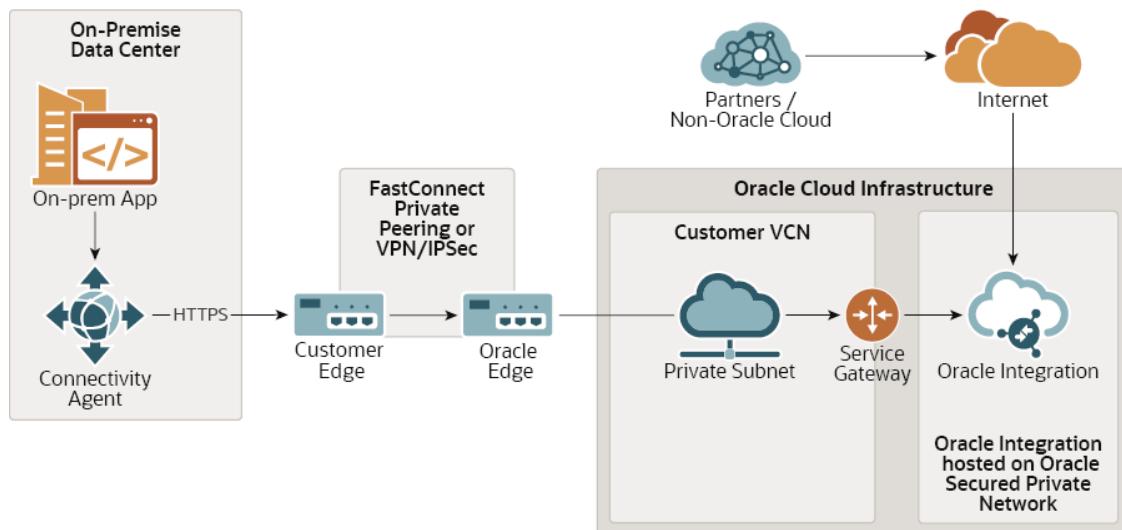
Connectivity Agent Deployed in Private or On-Premises Network

Install the connectivity agent in your private (on-premises) network, and set up a private connection between your network and VCN using FastConnect private peering or VPN. In addition, configure a service gateway to route the traffic from the VCN to Oracle Integration.

To use FastConnect, you must first subscribe to FastConnect with the private peering option. See [FastConnect](#). The FastConnect link contains the private IP addresses of the VCN.

If you want to use VPN, see [VPN Connect](#).

To configure a service gateway, see [FastConnect and VPN with Oracle Integration Cloud \(OIC\)](#).



Connectivity Agent Deployed in a VCN

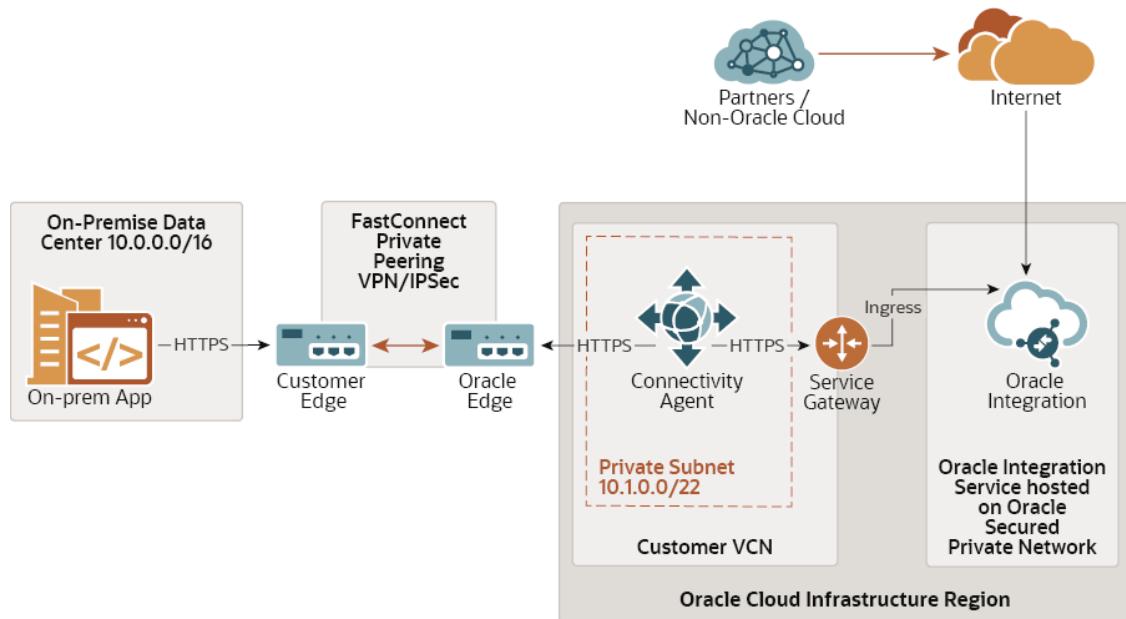
Install the connectivity agent in your VCN within Oracle Cloud Infrastructure, and set up a private connection between your network and the connectivity agent using FastConnect private peering or VPN. In addition, configure a service gateway to route the traffic from the VCN to Oracle Integration.

You can use this pattern if you have limited capacity or resource constraints on your data center.

For details on configuring a service gateway, see [FastConnect and VPN with Oracle Integration Cloud \(OIC\)](#).

Note:

The connectivity agent deployed in a VCN can also be used to access resources deployed in an Oracle Cloud Infrastructure VCN.



About Connecting to Private Resources

You can configure access to private resources available in your virtual cloud network (VCN) in Oracle Integration.

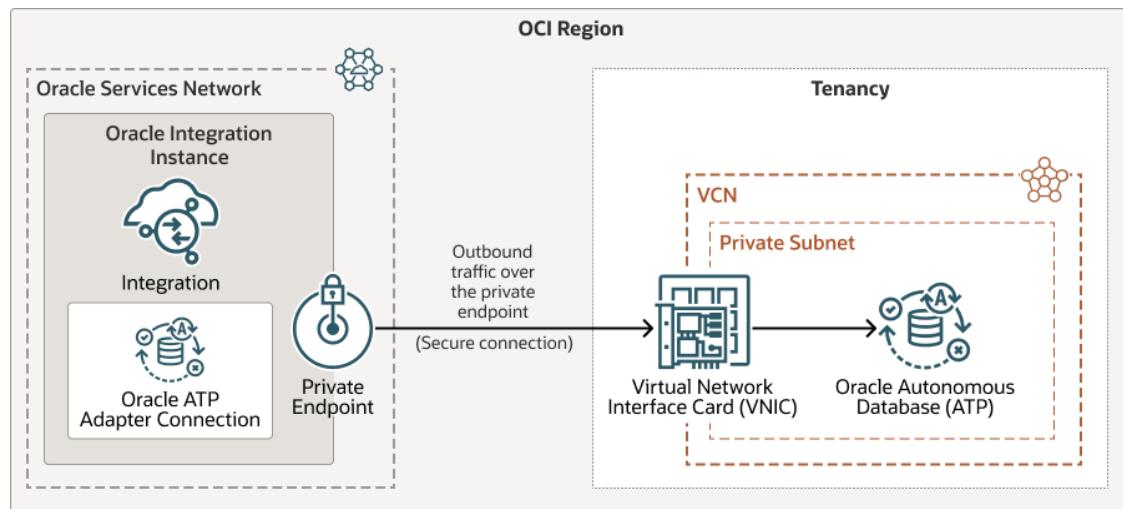
Outbound Private Connection Pattern

An outbound private connection pattern uses a private endpoint to connect to private resources that are in your virtual cloud network (VCN).

Create a virtual cloud network (VCN) and subnet, and add private resources to the subnet. Next, configure a private endpoint.

Outbound traffic from Oracle Integration goes through a private channel that is set up within the Oracle Cloud and connects to the private resources in the subnet. The traffic never goes through the public internet.

To get started, see Connect to Private Resources in *Provisioning and Administering Oracle Integration 3*.



Adapters that Support Connecting to Private Endpoints

You can configure access to private endpoints in Oracle Integration with the following adapters. You select the **Private endpoint** option in the **Access type** section of the Connections page.

- Apache Kafka Adapter
- AS2 Adapter
- eBay Marketplace Adapter
- FTP Adapter
- GraphQL Adapter
- HubSpot Adapter
- Jira Adapter
- Magento Commerce Cloud Adapter
- Marketo Adapter
- MS Dynamics 365 CRM Adapter
- OData Adapter
- Oracle Advanced Queuing (AQ) Adapter
- Oracle Autonomous Data Warehouse Adapter
- Oracle Autonomous Transaction Processing Adapter
- Oracle Cloud Infrastructure Streaming Service Adapter
- Oracle CPQ Adapter
- Oracle CX Sales and B2B Service Adapter
- Oracle Database Cloud Service Adapter
- Oracle ERP Cloud Adapter
- Oracle Field Service Cloud Adapter
- Oracle HCM Cloud Adapter
- Oracle Intelligent Advisor Adapter
- Oracle Primavera Cloud Adapter

- Oracle Primavera P6 EPPM Adapter
- Oracle Service Cloud (RightNow) Adapter
- Oracle Utilities Adapter
- PayPal Adapter
- QuickBooks Adapter
- Oracle Responsys Adapter
- REST Adapter
- Salesforce Adapter
- Salesforce Commerce Cloud Adapter
- Salesforce REST Adapter (inbound and outbound directions)
- SAP Ariba Adapter
- SAP Commerce Cloud (Hybris) Adapter
- SAP Concur Adapter
- SAP SuccessFactors Adapter
- SAP S/4HANA Cloud Adapter
- ServiceNow Adapter
- Shopify Adapter
- Shopify GraphQL Adapter
- SOAP Adapter
- SugarCRM Adapter
- Workday Adapter
- Zendesk Adapter

Requirements for Connecting to a Private Endpoint

When configuring access to a private endpoints in Oracle Integration, keep in mind several requirements.

- To connect to private endpoints, you must complete prerequisite tasks in the Oracle Cloud Infrastructure Console. Failure to do so results in errors when testing the connection. See Connect to Private Resources in *Provisioning and Administering Oracle Integration 3*.
- When configuring an adapter on the Connections page to connect to endpoints using a private network, specify the fully-qualified domain name (FQDN) and *not* the IP address. If you enter an IP address, validation fails when you click **Test**.
- IPSec tunneling and Fastconnect are not supported for use with private endpoints.

Integration Concepts

In every integration, you define information about the applications you are integrating and specify the source and destination of the data that the integration moves. Additionally, you

define the logic that controls the flow of data along with the mapping of data between different applications.

Topics:

- [About Integrations](#)
- [About Connections](#)
- [Available Adapters for Connections](#)
- [Understand Trigger and Invoke Connections](#)
- [About Mappings](#)
- [About Lookups](#)
- [About Packages](#)
- [About Business Identifiers for Tracking Fields in Messages](#)
- [About Error Management](#)

About Integrations

Integrations improve the efficiency and effectiveness of your business by sending data to applications and getting data from applications, seamlessly and without human intervention. Consider all the manual interventions that keep your business processes flowing as expected. An integration removes that manual work, thereby simplifying and streamlining your operations.

How Integrations Work

An integration moves data from one application to another. For the integration to pull data from one application and push the data to another application, the two applications' REST APIs must communicate with each other. An integration allows this communication to occur.

Moving Data, One Step at a Time

To understand an integration, think of an assembly line where an unfinished product moves from station to station in a specific order. Each station has special tools that build the product.

Of course, an integration is different from an assembly line. While an assembly line builds a product, an integration moves data between applications. However, both an assembly line and an integration follow a series of steps in a specific order. An integration follows these steps so that it can move data. You specify the steps and their order when you build the integration.

Related Components

In every integration, you specify how to communicate to applications using connections. Additionally, you specify how data is moved between applications by creating one or more mappings. When an integration needs to relate different terms used by different applications for the same item, such as a numeric country code used in one application and an alphabetic country code is used in another, you can use a lookup to relate the values.

See [About Connections](#), [About Mappings](#), and [About Lookups](#).

Save Time with Recipes and Accelerators

Oracle offers several time-saving features for integrations, including recipes and accelerators. Recipes and accelerators are custom built for specific business needs and include everything you need to build an integration, including mappings and lookups. Before you start designing

an integration, review the library of recipes and accelerators and see if they can help you jump start your integration work. See [Recipes and Accelerators](#) on the Oracle Help Center.

About Connections

A connection tells your integration how to connect to an application and what to do after connecting. Most integrations contain at least one connection.

Key Points for Connections

1. Every connection is based on an adapter.
2. Two types of connections are available:
 - Trigger connections run the integration and pull data from a source application.
 - Invoke connections send data to a target application.
3. To connect to an application from an integration, you must perform the following steps:
 - a. Create a connection to an application.
 - b. Add the connection to an integration.

Keep reading for more details.

Connections Are Based on Adapters

When you create a connection, you choose the adapter to base the connection on. An adapter provides an interface so that an integration can connect with an application.

Oracle Integration provides an always growing library of application-specific adapters, such as Oracle CX Sales and B2B Service Adapter and the Oracle Service Cloud (RightNow) Adapter.

Additionally, some adapters let you connect to just about any product's REST APIs, even if an adapter isn't available for the product. These adapters are the most popular and include:

- REST Adapter
- SOAP Adapter
- FTP Adapter

For a list of all available adapters, see [Adapters](#) on the Oracle Help Center.

Each Adapter Requires Different Information

After choosing an adapter, you specify the information that the adapter requires. For example, for a connection based on the Oracle Service Cloud (RightNow) Adapter, you must specify the WSDL URL, security policy, and security credentials to connect to it.

Tip:

If the connection is based on the REST Adapter, you can test the integration without having to deploy it. See [Test REST Adapter Trigger Connection-Based Integrations in Projects](#).

Two-Step Process for Connections

An integration sends or receives data for one or more applications using connections. For example, a connection based on the Oracle Service Cloud (RightNow) Adapter can send and receive data from Oracle Service Cloud (RightNow). Adding a connection to an integration is a multi-step process:

1. Create the connection.

Specify the adapter and the role of the connection, such as trigger connection, invoke connection, or trigger and invoke connection. Depending on the adapter and role you choose, you must provide additional information. For example, if you choose a database adapter, enter the IP address for the database, the user name and password for connecting to the database, and the security policy for securing the data. The role that you choose determines the security policies that are available for the connection.

After creating the connection, test it to ensure that it connects to the application.

See [Add and Share a Connection Across a Project](#) and [Understand Trigger and Invoke Connections](#).

2. Configure the connection for use in an integration.

You can use each connection in as many integrations as you want, and you can use the connection to move different data.

For example, consider ten integrations that send data to Oracle ERP Cloud. All the integrations use the same invoke connection that is based on the Oracle ERP Cloud Adapter. However, each connection sends different data. When you add a connection to an integration, you choose the data to move in the source application and specify how it moves into the target application. If you need to process the data, you specify that information, too. Examples of processing tasks include the following:

- Define the type of operations to perform on the data.
- Specify the business objects and fields against which to perform the operations.
- Define the required schemas.

See [Configure Connection Properties and Security Properties](#).

Typical Components of Integrations

Most integrations contain at least one of each of the following components:

• Trigger connection

A trigger connection can be a scheduled trigger that is based on a schedule, or it can be a trigger that is based on a connection you define. A trigger connection does two things: It starts running the integration, and it pulls data from a source application.

For example, consider an integration that automates the creation of an order when a customer approves an opportunity, such as a quote. The integration connects to Salesforce using the Salesforce Adapter. The connection is a trigger connection, and it pulls opportunity data from Salesforce.

Every integration must contain a trigger connection.

• Invoke connection

Create an invoke connection, also called a target connection, when a target application must receive data from the source application using the integration.

Most integrations contain at least one invoke connection. For example, if your integration obtains the lowest price for an item from suppliers, it uses one invoke connection for each supplier to receive this data.

In the Salesforce opportunity-to-order scenario, Salesforce is both the source application and the target application. The integration collects opportunity information from Salesforce and uses it to automatically generate an order in Salesforce.

- **Mapping between the trigger connection and invoke connection**

The mapping between the connections specifies how the integration transfers data from the trigger (source) application to the invoke (target) application.

The integration uses the connections to send request messages to applications and to receive response messages from applications. You map the request and response messages as required by your use case. For example, you can map a request message to a response message, or map two request messages.

Most integrations contain at least one mapping.

Available Adapters for Connections

You can create connections using application-specific adapters. The list of adapters is always growing. Additionally, you can use the REST, SOAP, and FTP adapters to connect to just about any product, even if it doesn't yet have an adapter. You can also build your own adapter with the Rapid Adapter Builder.

- [List of Available Adapters](#)
- [Build Your Own Adapter with the Rapid Adapter Builder](#)

List of Available Adapters

Adapter	For Information
Adobe Sign Adapter	Using the Adobe Sign Adapter with Oracle Integration 3
Apache Kafka Adapter	Using the Apache Kafka Adapter with Oracle Integration 3
AS2 Adapter	Using the AS2 Adapter with Oracle Integration 3
AS4 Adapter	Using the AS4 Adapter with Oracle Integration 3
Amazon Simple Notification Service (SNS) Adapter	Using the Amazon Simple Notification Service (SNS) Adapter with Oracle Integration 3
Amazon Simple Queue Service (SQS) Adapter	Using the Amazon Simple Queue Service (SQS) Adapter with Oracle Integration 3
Amazon Simple Storage Service (S3) Adapter	Using the Amazon Simple Storage Service (S3) Adapter with Oracle Integration 3
<i>Using the ArcGIS (ESRI) Adapter with Oracle Integration 3</i>	Using the ArcGIS (ESRI) Adapter with Oracle Integration 3
Azure Active Directory Adapter	Using the Azure Active Directory Adapter with Oracle Integration 3
Azure Event Grid Adapter	Using the Azure Event Grid Adapter with Oracle Integration 3
Azure Service Bus Adapter	Using the Azure Service Bus Adapter with Oracle Integration 3
Azure Storage Adapter	Using the Azure Storage Adapter with Oracle Integration 3
Box Adapter	Using the Box Adapter with Oracle Integration 3

Adapter	For Information
Confluent Adapter	<i>Using the Confluent Adapter with Oracle Integration 3</i>
Coupa Procurement Adapter	<i>Using the Oracle Procurement Adapter for Oracle Integration 3</i>
Coupa Sourcing Adapter	<i>Using the Coupa Sourcing Adapter with Oracle Integration 3</i>
DocuSign Adapter	<i>Using the DocuSign Adapter with Oracle Integration 3</i>
eBay Marketplace Adapter	<i>Using the eBay Marketplace Adapter with Oracle Integration 3</i>
Eventbrite Adapter	<i>Using the Eventbrite Adapter with Oracle Integration 3</i>
Facebook Adapter	<i>Using the Facebook Adapter with Oracle Integration 3</i>
FHIR Adapter	<i>Using the FHIR Adapter with Oracle Integration 3</i>
File Adapter	<i>Using the File Adapter with Oracle Integration 3</i>
FTP Adapter	<i>Using the FTP Adapter with Oracle Integration 3</i>
GCP Pub Sub Adapter	<i>Using the GCP Pub Sub Adapter with Oracle Integration 3</i>
GCP Storage Adapter	<i>Using the GCP Storage Adapter with Oracle Integration 3</i>
GitHub Adapter	<i>Using the GitHub Adapter with Oracle Integration 3</i>
GraphQL Adapter	<i>Using the GraphQL Adapter with Oracle Integration 3</i>
Google Gmail Adapter	<i>Using the Google Gmail Adapter with Oracle Integration 3</i>
Google Calendar Adapter	<i>Using the Google Calendar Adapter with Oracle Integration 3</i>
Google Tasks Adapter	<i>Using the Google Tasks Adapter with Oracle Integration 3</i>
HubSpot Adapter	<i>Using the HubSpot Adapter with Oracle Integration 3</i>
IBM DB2 Adapter	<i>Using the IBM DB2 Adapter with Oracle Integration 3</i>
IBM MQ Series JMS Adapter	<i>Using the IBM MQ Series JMS Adapter with Oracle Integration 3</i>
Jira Adapter	<i>Using the Jira Adapter with Oracle Integration 3</i>
LinkedIn Adapter	<i>Using the LinkedIn Adapter with Oracle Integration 3</i>
Magento Commerce Cloud Adapter	<i>Using the Magento Commerce Cloud Adapter with Oracle Integration 3</i>
Mailchimp Adapter	<i>Using the Mailchimp Adapter with Oracle Integration 3</i>
Marketo Adapter	<i>Using the Marketo Adapter with Oracle Integration 3</i>
Microsoft Office 365 Calendar Adapter	<i>Using the Microsoft Office 365 Calendar Adapter with Oracle Integration 3</i>
Microsoft Office 365 People Adapter	<i>Using the Microsoft Office 365 People Adapter with Oracle Integration 3</i>
Microsoft Office 365 Outlook Adapter	<i>Using the Microsoft Office 365 Outlook Adapter with Oracle Integration 3</i>

Adapter	For Information
Microsoft SQL Server Adapter	<i>Using the Microsoft SQL Server Adapter with Oracle Integration 3</i>
MLLP Adapter	<i>Using the MLLP Adapter with Oracle Integration 3</i>
MS Dynamics 365 CRM Adapter	<i>Using the MS Dynamics 365 CRM Adapter with Oracle Integration 3</i>
MySQL Adapter	<i>Using the MySQL Adapter with Oracle Integration 3</i>
Netezza Adapter	<i>Using the Netezza Adapter with Oracle Integration 3</i>
OData Adapter	<i>Using the OData Adapter with Oracle Integration 3</i>
Oracle Aconex Cloud Adapter	<i>Using the Oracle Aconex Cloud Adapter with Oracle Integration 3</i>
Oracle Advanced Queuing (AQ) Adapter	<i>Using the Oracle Advanced Queuing (AQ) Adapter with Oracle Integration 3</i>
Oracle Autonomous Data Warehouse Adapter	<i>Using the Oracle Autonomous Data Warehouse Adapter with Oracle Integration 3</i>
Oracle Autonomous Transaction Processing Adapter	<i>Using the Oracle Autonomous Transaction Processing Adapter with Oracle Integration 3</i>
Oracle Cloud Infrastructure Streaming Service Adapter	<i>Using the Oracle Cloud Infrastructure Streaming Service Adapter with Oracle Integration 3</i>
Oracle Commerce Cloud Adapter	<i>Using the Oracle Commerce Cloud Adapter with Oracle Integration 3</i>
Oracle CPQ Adapter	<i>Using the Oracle CPQ Adapter with Oracle Integration 3</i>
Oracle Database Adapter	<i>Using the Oracle Database Adapter with Oracle Integration 3</i>
Oracle Database Cloud Service Adapter	<i>Using the Oracle Database Cloud Service Adapter with Oracle Integration 3</i>
Oracle E-Business Suite Adapter	<i>Using the Oracle E-Business Suite Adapter with Oracle Integration 3</i>
Oracle Eloqua Cloud Adapter	<i>Using the Oracle Eloqua Cloud Adapter with Oracle Integration 3</i>
Oracle CX Sales and B2B Service Adapter	<i>Using the Oracle CX Sales and B2B Service Adapter with Oracle Integration 3</i>
Oracle Enterprise Performance Management Cloud Adapter	<i>Using the Oracle Enterprise Performance Management Cloud Adapter with Oracle Integration 3</i>
Oracle ERP Cloud Adapter	<i>Using the Oracle ERP Cloud Adapter with Oracle Integration 3</i>
Oracle Field Service Cloud Adapter	<i>Using the Oracle Field Service Cloud Adapter with Oracle Integration 3</i>
Oracle HCM Cloud Adapter	<i>Using the Oracle HCM Cloud Adapter with Oracle Integration 3</i>
Oracle Hospitality Adapter	<i>Using the Oracle Hospitality Adapter with Oracle Integration 3</i>
Oracle Intelligent Advisor Adapter	<i>Using the Oracle Intelligent Advisor Adapter with Oracle Integration 3</i>
Oracle Intelligent Track and Trace Adapter	<i>Using the Oracle Intelligent Track and Trace Adapter with Oracle Integration 3</i>
Oracle JD Edwards EnterpriseOne Adapter	<i>Using the Oracle JD Edwards EnterpriseOne Adapter with Oracle Integration 3</i>
Oracle Logistics Adapter	<i>Using the Oracle Logistics Adapter with Oracle Integration 3</i>

Adapter	For Information
Oracle NetSuite Adapter	<i>Using the Oracle NetSuite Adapter with Oracle Integration 3</i>
Oracle Primavera Cloud Adapter	<i>Using the Oracle Primavera Cloud Adapter with Oracle Integration 3</i>
Oracle Primavera Cloud Data Service Adapter	<i>Using the Oracle Primavera Cloud Data Service Adapter with Oracle Integration 3</i>
Oracle Primavera P6 EPPM Adapter	<i>Using the Oracle Primavera P6 EPPM Adapter with Oracle Integration 3</i>
Oracle Responsys Adapter	<i>Using the Oracle Responsys Adapter with Oracle Integration 3</i>
Oracle Service Cloud (RightNow) Adapter	<i>Using the Oracle Service Cloud (RightNow) Adapter with Oracle Integration 3</i>
Oracle Siebel Adapter	<i>Using the Oracle Siebel Adapter with Oracle Integration 3</i>
Oracle SOA Suite Adapter	<i>Using the Oracle SOA Suite Adapter with Oracle Integration 3</i>
Oracle Taleo Enterprise Edition Adapter	<i>Using the Oracle Taleo Enterprise Edition Adapter with Oracle Integration 3</i>
Oracle Unity Adapter	<i>Using the Oracle Unity Adapter with Oracle Integration</i>
Oracle Utilities Adapter	<i>Using the Oracle Utilities Adapter with Oracle Integration 3</i>
Oracle WebLogic JMS Adapter	<i>Using the Oracle WebLogic JMS Adapter with Oracle Integration 3</i>
PayPal Adapter	<i>Using the PayPal Adapter with Oracle Integration 3</i>
PostgreSQL Adapter	<i>Using the PostgreSQL Adapter with Oracle Integration 3</i>
Primavera Unifier Adapter	<i>Using the Oracle Primavera Unifier Adapter with Oracle Integration 3</i>
REST Adapter	<i>Using the REST Adapter with Oracle Integration 3</i>
RosettaNet Adapter	<i>Using the RosettaNet Adapter with Oracle Integration 3</i>
Salesforce Adapter	<i>Using the Salesforce Adapter with Oracle Integration 3</i>
Salesforce Commerce Cloud Adapter	<i>Using the Salesforce Commerce Cloud Adapter with Oracle Integration 3</i>
Salesforce Marketing Cloud Adapter	<i>Using the Salesforce Marketing Cloud Adapter with Oracle Integration 3</i>
Salesforce REST Adapter	<i>Using the Salesforce REST Adapter with Oracle Integration 3</i>
SAP Adapter	<i>Using the SAP Adapter with Oracle Integration 3</i>
SAP Ariba Adapter	<i>Using the SAP Ariba Adapter with Oracle Integration 3</i>
SAP ASE (Sybase) Adapter	<i>Using the SAP ASE (Sybase) Adapter with Oracle Integration 3</i>
SAP Commerce Cloud (Hybris) Adapter	<i>Using the SAP Commerce Cloud (Hybris) Adapter with Oracle Integration 3</i>
SAP Concur Adapter	<i>Using the SAP Concur Adapter with Oracle Integration 3</i>
SAP SuccessFactors Adapter	<i>Using the SAP SuccessFactors Adapter with Oracle Integration 3</i>

Adapter	For Information
SAP S/4HANA Cloud Adapter	<i>Using the SAP S/4HANA Cloud Adapter with Oracle Integration 3</i>
ServiceNow Adapter	<i>Using the ServiceNow Adapter with Oracle Integration 3</i>
Shopify Adapter	<i>Using the Shopify Adapter with Oracle Integration 3</i>
Shopify GraphQL Adapter	<i>Using the Shopify GraphQL Adapter with Oracle Integration 3</i>
Slack Adapter	<i>Using the Slack Adapter with Oracle Integration 3</i>
Snowflake Adapter	<i>Using the Snowflake Adapter with Oracle Integration 3</i>
SOAP Adapter	<i>Using the SOAP Adapter with Oracle Integration 3</i>
SugarCRM Adapter	<i>Using the SugarCRM Adapter with Oracle Integration 3</i>
SurveyMonkey Adapter	<i>Using the SurveyMonkey Adapter with Oracle Integration 3</i>
Trello Adapter	<i>Using the Trello Adapter with Oracle Integration 3</i>
Twilio Adapter	<i>Using the Twilio Adapter with Oracle Integration 3</i>
Twitter Adapter	<i>Using the Twitter Adapter with Oracle Integration 3</i>
Workday Adapter	<i>Using the Workday Adapter with Oracle Integration 3</i>
Zendesk Adapter	<i>Using the Zendesk Adapter with Oracle Integration 3</i>
Zuora Adapter	<i>Using the Zuora Adapter with Oracle Integration 3</i>

Build Your Own Adapter with the Rapid Adapter Builder

Not finding the adapter you need in the list? You can build your own adapter with the Rapid Adapter Builder.

The Rapid Adapter Builder transforms the adapter experience in Oracle Integration by allowing you to build an adapter for any application that exposes REST APIs.

Oracle provides a Visual Studio Code extension for the Rapid Adapter Builder. Use the extension to develop, validate, and publish an adapter to an Oracle Integration instance.

See Learn About the Rapid Adapter Builder in Oracle Integration in *Using the Rapid Adapter Builder with Oracle Integration 3*.

Understand Trigger and Invoke Connections

When you design an integration, you drag trigger (source) and invoke (target) adapter connections into the integration.

The information required to connect to the application is already defined in the connection. However, you still must specify certain information, such as business object, operation, or other elements to use for the request and response and how to process the data. You invoke the Adapter Endpoint Configuration Wizard to guide you through configuration of this adapter connection.

For example, you design an integration that synchronizes customer information between an Oracle Service Cloud (RightNow) application (the trigger) and an Oracle CX Sales and B2B Service application (the invoke) in real time. Whenever an Organization object is created in Oracle Service Cloud (RightNow), an AccountService object is created in Oracle CX Sales and

B2B Service in real time. To achieve this scenario, you define both an Oracle Service Cloud (RightNow) Adapter as a trigger connection and an Oracle CX Sales and B2B Service Adapter as an invoke connection in the Adapter Endpoint Configuration Wizard.

 **Note:**

A trigger adapter connection is not required in schedule integrations. The schedule acts as the trigger in schedule integrations.

Trigger Connections

You can define the following adapter connections as triggers in an integration. The trigger (source) connection sends requests from your inbound application to Oracle Integration. The following guides describe the pages of the Adapter Endpoint Configuration Wizard for each adapter connection.

Adapter	For Information
Amazon Simple Notification Service (SNS) Adapter	<i>Using the Amazon Simple Notification Service (SNS) Adapter with Oracle Integration 3</i>
Apache Kafka Adapter	<i>Using the Apache Kafka Adapter with Oracle Integration 3</i>
AS2 Adapter	<i>Using the AS2 Adapter with Oracle Integration 3</i>
Azure Active Directory Adapter	<i>Using the Azure Active Directory Adapter with Oracle Integration 3</i>
Azure Event Grid Adapter	<i>Using the Azure Event Grid Adapter with Oracle Integration 3</i>
Azure Service Bus Adapter	<i>Using the Azure Service Bus Adapter with Oracle Integration 3</i>
Box Adapter	<i>Using the Box Adapter with Oracle Integration 3</i>
Confluent Adapter	<i>Using the Confluent Adapter with Oracle Integration 3</i>
File Adapter	<i>Using the File Adapter with Oracle Integration 3</i>
GCP Pub Sub Adapter	<i>Using the GCP Pub Sub Adapter with Oracle Integration 3</i>
GitHub Adapter	<i>Using the GitHub Adapter with Oracle Integration 3</i>
HubSpot Adapter	<i>Using the HubSpot Adapter with Oracle Integration 3</i>
IBM DB2 Adapter	<i>Using the IBM DB2 Adapter with Oracle Integration 3</i>
IBM MQ Series JMS Adapter	<i>Using the IBM MQ Series JMS Adapter with Oracle Integration 3</i>
Jira Adapter	<i>Using the Jira Adapter with Oracle Integration 3</i>
Microsoft SQL Server Adapter	<i>Using the Microsoft SQL Server Adapter with Oracle Integration 3</i>
MLLP Adapter	<i>Using the MLLP Adapter with Oracle Integration 3</i>
MySQL Adapter	<i>Using the MySQL Adapter with Oracle Integration 3</i>
Netezza Adapter	<i>Using the Netezza Adapter with Oracle Integration 3</i>
Oracle Advanced Queuing (AQ) Adapter	<i>Using the Oracle Advanced Queuing (AQ) Adapter with Oracle Integration 3</i>

Adapter	For Information
Oracle Autonomous Data Warehouse Adapter	<i>Using the Oracle Autonomous Data Warehouse Adapter with Oracle Integration 3</i>
Oracle Autonomous Transaction Processing Adapter	<i>Using the Oracle Autonomous Transaction Processing Adapter with Oracle Integration 3</i>
Oracle Cloud Infrastructure Streaming Service Adapter	<i>Using the Oracle Cloud Infrastructure Streaming Service Adapter with Oracle Integration 3</i>
Oracle Commerce Cloud Adapter	<i>Using the Oracle Commerce Cloud Adapter with Oracle Integration 3</i>
Oracle CPQ Adapter	<i>Using the Oracle CPQ Adapter with Oracle Integration 3</i>
Oracle Database Adapter	<i>Using the Oracle Database Adapter with Oracle Integration 3</i>
Oracle Database Cloud Service Adapter	<i>Using the Oracle Database Cloud Service Adapter with Oracle Integration 3</i>
Oracle E-Business Suite Adapter	<i>Using Oracle E-Business Suite Adapter with Oracle Integration 3</i>
Oracle Eloqua Cloud Adapter	<i>Using the Oracle Eloqua Cloud Adapter with Oracle Integration 3</i>
Oracle CX Sales and B2B Service Adapter	<i>Using the Oracle CX Sales and B2B Service Adapter with Oracle Integration 3</i>
Oracle ERP Cloud Adapter	<i>Using the Oracle ERP Cloud Adapter with Oracle Integration 3</i>
Oracle Field Service Cloud Adapter	<i>Using the Oracle Field Service Cloud Adapter with Oracle Integration 3</i>
Oracle HCM Cloud Adapter	<i>Using the Oracle HCM Cloud Adapter with Oracle Integration 3</i>
Oracle Hospitality Adapter	<i>Using the Oracle Hospitality Adapter with Oracle Integration 3</i>
Oracle JD Edwards EnterpriseOne Adapter	<i>Using the JD Edwards EnterpriseOne Adapter with Oracle Integration 3</i>
Oracle Intelligent Advisor Adapter	<i>Using the Oracle Intelligent Advisor Adapter with Oracle Integration 3</i>
Oracle Logistics Adapter	<i>Oracle Logistics Adapter with Oracle Integration 3</i>
Oracle Service Cloud (RightNow) Adapter	<i>Using the Oracle Service Cloud (RightNow) Adapter with Oracle Integration 3</i>
Oracle Siebel Adapter	<i>Using the Oracle Siebel Adapter with Oracle Integration 3</i>
Oracle WebLogic JMS Adapter	<i>Using the Oracle WebLogic JMS Adapter with Oracle Integration 3</i>
Oracle Utilities Adapter	<i>Using the Oracle Utilities Adapter with Oracle Integration 3</i>
PayPal Adapter	<i>Using the PayPal Adapter with Oracle Integration 3</i>
PostgreSQL Adapter	<i>Using the PostgreSQL Adapter with Oracle Integration 3</i>
Primavera Unifier Adapter	<i>Using the Primavera Unifier Adapter with Oracle Integration 3</i>
REST Adapter	<i>Using the REST Adapter with Oracle Integration 3</i>
RosettaNet Adapter	<i>Using the RosettaNet Adapter with Oracle Integration 3</i>
Salesforce Adapter	<i>Using the Salesforce Adapter with Oracle Integration 3</i>

Adapter	For Information
SAP Adapter	<i>Using the SAP Adapter with Oracle Integration 3</i>
Salesforce Marketing Cloud Adapter	<i>Using the Salesforce Marketing Cloud Adapter with Oracle Integration 3</i>
Salesforce REST Adapter	<i>Using the Salesforce REST Adapter with Oracle Integration 3</i>
SAP Adapter	<i>Using the SAP Adapter with Oracle Integration 3</i>
SAP Ariba Adapter	<i>Using the SAP Ariba Adapter with Oracle Integration 3</i>
SAP ASE (Sybase) Adapter	<i>Using the SAP ASE (Sybase) Adapter for Oracle Integration 3</i>
SAP SuccessFactors Adapter	<i>Using the SAP SuccessFactors Adapter with Oracle Integration 3</i>
ServiceNow Adapter	<i>Using the ServiceNow Adapter with Oracle Integration 3</i>
Shopify Adapter	<i>Using the Shopify Adapter with Oracle Integration 3</i>
Shopify GraphQL Adapter	<i>Using the Shopify GraphQL Adapter with Oracle Integration 3</i>
SOAP Adapter	<i>Using the SOAP Adapter with Oracle Integration 3</i>
Zendesk Adapter	<i>Using the Zendesk Adapter with Oracle Integration 3</i>
Zuora Adapter	<i>Using the Zuora Adapter with Oracle Integration 3</i>

Invoke Connections

You can define the following adapter connections as invokes in an integration. Oracle Integration sends requests or information to the invoke (target) connection. The following guides describe the pages of the Adapter Endpoint Configuration Wizard for each adapter connection.

Adapter	For Information
Adobe Sign Adapter	<i>Using the Adobe Sign Adapter with Oracle Integration 3</i>
Apache Kafka Adapter	<i>Using the Apache Kafka Adapter with Oracle Integration 3</i>
Using the ArcGIS (ESRI) Adapter with Oracle Integration 3	<i>Using the ArcGIS (ESRI) Adapter with Oracle Integration 3</i>
AS2 Adapter	<i>Using the AS2 Adapter with Oracle Integration 3</i>
AS4 Adapter	<i>Using the AS4 Adapter with Oracle Integration 3</i>
Amazon Simple Notification Service (SNS) Adapter	<i>Using the Amazon Simple Notification Service (SNS) Adapter with Oracle Integration 3</i>
Amazon Simple Queue Service (SQS) Adapter	<i>Using the Amazon Simple Queue Service (SQS) Adapter with Oracle Integration 3</i>
Amazon Simple Storage Service (S3) Adapter	<i>Using the Amazon Simple Storage Service (S3) Adapter with Oracle Integration 3</i>
Azure Event Grid Adapter	<i>Using the Azure Event Grid Adapter with Oracle Integration 3</i>
Azure Service Bus Adapter	<i>Using the Azure Service Bus Adapter with Oracle Integration 3</i>
Azure Active Directory Adapter	<i>Using the Azure Active Directory Adapter with Oracle Integration 3</i>

Adapter	For Information
Azure Storage Adapter	<i>Using the Azure Storage Adapter with Oracle Integration 3</i>
Box Adapter	<i>Using the Box Adapter with Oracle Integration 3</i>
Confluent Adapter	<i>Using the Confluent Adapter with Oracle Integration 3</i>
Coupa Procurement Adapter	<i>Using the Coupa Procurement Adapter with Oracle Integration 3</i>
Coupa Sourcing Adapter	<i>Using the Coupa Sourcing Adapter with Oracle Integration 3</i>
DocuSign Adapter	<i>Using the DocuSign Adapter with Oracle Integration 3</i>
eBay Marketplace Adapter	<i>Using the eBay Marketplace Adapter with Oracle Integration 3</i>
Eventbrite Adapter	<i>Using the Eventbrite Adapter with Oracle Integration 3</i>
Facebook Adapter	<i>Using the Facebook Adapter with Oracle Integration 3</i>
FHIR Adapter	<i>Using the FHIR Adapter with Oracle Integration 3</i>
File Adapter	<i>Using the File Adapter with Oracle Integration 3</i>
FTP Adapter	<i>Using the FTP Adapter with Oracle Integration 3</i>
GCP Pub Sub Adapter	<i>Using the GCP Pub Sub Adapter with Oracle Integration 3</i>
GCP Storage Adapter	<i>Using the GCP Storage Adapter with Oracle Integration 3</i>
GitHub Adapter	<i>Using the GitHub Adapter with Oracle Integration 3</i>
GraphQL Adapter	<i>Using the GraphQL Adapter with Oracle Integration 3</i>
Google Gmail Adapter	<i>Using the Google Gmail Adapter with Oracle Integration 3</i>
Google Calendar Adapter	<i>Using the Google Calendar Adapter with Oracle Integration 3</i>
Google Tasks Adapter	<i>Using the Google Tasks Adapter with Oracle Integration 3</i>
HubSpot Adapter	<i>Using the HubSpot Adapter with Oracle Integration 3</i>
IBM DB2 Adapter	<i>Using the IBM DB2 Adapter with Oracle Integration 3</i>
IBM MQ Series JMS Adapter	<i>Using the IBM MQ Series JMS Adapter with Oracle Integration 3</i>
Jira Adapter	<i>Using the Jira Adapter with Oracle Integration 3</i>
LinkedIn Adapter	<i>Using the LinkedIn Adapter with Oracle Integration 3</i>
Magento Commerce Cloud Adapter	<i>Using the Magento Commerce Cloud Adapter with Oracle Integration 3</i>
Mailchimp Adapter	<i>Using the Mailchimp Adapter with Oracle Integration 3</i>
Marketo Adapter	<i>Using the Marketo Adapter with Oracle Integration 3</i>
Microsoft Office 365 Calendar Adapter	<i>Using the Microsoft Office 365 Calendar Adapter with Oracle Integration 3</i>

Adapter	For Information
Microsoft Office 365 People Adapter	<i>Using the Microsoft Office 365 People Adapter with Oracle Integration 3</i>
Microsoft Office 365 Outlook Adapter	<i>Using the Microsoft Office 365 Outlook Adapter with Oracle Integration 3</i>
Microsoft SQL Server Adapter	<i>Using the Microsoft SQL Server Adapter with Oracle Integration 3</i>
MLLP Adapter	<i>Using the MLLP Adapter with Oracle Integration 3</i>
MS Dynamics 365 CRM Adapter	<i>Using the MS Dynamics 365 CRM Adapter with Oracle Integration 3</i>
MySQL Adapter	<i>Using the MySQL Adapter with Oracle Integration 3</i>
Netezza Adapter	<i>Using the Netezza Adapter with Oracle Integration 3</i>
OData Adapter	<i>Using the OData Adapter with Oracle Integration 3</i>
Oracle Aconex Cloud Adapter	<i>Using the Oracle Aconex Cloud Adapter with Oracle Integration 3</i>
Oracle Advanced Queuing (AQ) Adapter	<i>Using the Oracle Advanced Queuing (AQ) Adapter with Oracle Integration 3</i>
Oracle Autonomous Data Warehouse Adapter	<i>Using the Oracle Autonomous Data Warehouse Adapter with Oracle Integration 3</i>
Oracle Autonomous Transaction Processing Adapter	<i>Using the Oracle Autonomous Transaction Processing Adapter with Oracle Integration 3</i>
Oracle Cloud Infrastructure Streaming Service Adapter	<i>Using the Oracle Cloud Infrastructure Streaming Service Adapter with Oracle Integration 3</i>
Oracle Commerce Cloud Adapter	<i>Using the Oracle Commerce Cloud Adapter with Oracle Integration 3</i>
Oracle CPQ Adapter	<i>Using the Oracle CPQ Adapter with Oracle Integration 3</i>
Oracle Database Adapter	<i>Using the Oracle Database Adapter with Oracle Integration 3</i>
Oracle Eloqua Cloud Adapter	<i>Using the Oracle Eloqua Cloud Adapter with Oracle Integration 3</i>
Oracle CX Sales and B2B Service Adapter	<i>Using the Oracle CX Sales and B2B Service Adapter with Oracle Integration 3</i>
Oracle Database Cloud Service Adapter	<i>Using the Oracle Database Cloud Service Adapter with Oracle Integration 3</i>
Oracle E-Business Suite Adapter	<i>Using Oracle E-Business Suite Adapter with Oracle Integration 3</i>
Oracle ERP Cloud Adapter	<i>Using the Oracle ERP Cloud Adapter with Oracle Integration 3</i>
Oracle Field Service Cloud Adapter	<i>Using the Oracle Field Service Cloud Adapter with Oracle Integration 3</i>
Oracle HCM Cloud Adapter	<i>Using the Oracle HCM Cloud Adapter with Oracle Integration 3</i>
Oracle Hospitality Adapter	<i>Using the Oracle Hospitality Adapter with Oracle Integration 3</i>
Oracle Intelligent Advisor Adapter	<i>Using the Oracle Intelligent Advisor Adapter with Oracle Integration 3</i>
Oracle Intelligent Track and Trace Adapter	<i>Using the Oracle Intelligent Track and Trace Adapter with Oracle Integration 3</i>
Oracle JD Edwards EnterpriseOne Adapter	<i>Using the JD Edwards EnterpriseOne Adapter with Oracle Integration 3</i>

Adapter	For Information
Oracle Logistics Adapter	<i>Using the Oracle Logistics Cloud Adapter with Oracle Integration 3</i>
Oracle NetSuite Adapter	<i>Using the Oracle NetSuite Adapter with Oracle Integration 3</i>
Oracle Primavera Cloud Adapter	<i>Using the Oracle Primavera Cloud Adapter with Oracle Integration 3</i>
Oracle Primavera Cloud Data Service Adapter	<i>Using the Oracle Primavera Cloud Data Service Adapter with Oracle Integration 3</i>
Oracle Primavera P6 EPPM Adapter	<i>Using the Oracle Primavera P6 EPPM Adapter with Oracle Integration 3</i>
Oracle Responsys Adapter	<i>Using the Oracle Responsys Adapter with Oracle Integration 3</i>
Oracle Service Cloud (RightNow) Adapter	<i>Using the Oracle Service Cloud (RightNow) Adapter with Oracle Integration 3</i>
Oracle Siebel Adapter	<i>Using the Oracle Siebel Adapter with Oracle Integration 3</i>
Oracle SOA Suite Adapter	<i>Using the Oracle SOA Suite Adapter with Oracle Integration 3</i>
Oracle Taleo Enterprise Edition Adapter	<i>Using the Oracle Taleo Enterprise Edition Adapter with Oracle Integration 3</i>
Oracle Unity Adapter	<i>Using the Oracle Unity Adapter with Oracle Integration</i>
Oracle Utilities Adapter	<i>Using the Oracle Utilities Adapter with Oracle Integration 3</i>
Oracle WebLogic JMS Adapter	<i>Using the Oracle WebLogic JMS Adapter with Oracle Integration 3</i>
PayPal Adapter	<i>Using the PayPal Adapter with Oracle Integration 3</i>
PostgreSQL Adapter	<i>Using the PostgreSQL Adapter with Oracle Integration 3</i>
Primavera Unifier Adapter	<i>Using the Primavera Unifier Adapter with Oracle Integration 3</i>
QuickBooks Adapter	<i>Using the QuickBooks Adapter with Oracle Integration 3</i>
REST Adapter	<i>Using the REST Adapter with Oracle Integration 3</i>
RosettaNet Adapter	<i>Using the RosettaNet Adapter with Oracle Integration 3</i>
Salesforce Adapter	<i>Using the Salesforce Adapter with Oracle Integration 3</i>
Salesforce Commerce Cloud Adapter	<i>Using the Salesforce Commerce Cloud Adapter with Oracle Integration 3</i>
Salesforce Marketing Cloud Adapter	<i>Using the Salesforce Marketing Cloud Adapter with Oracle Integration 3</i>
Salesforce REST Adapter	<i>Using the Salesforce REST Adapter with Oracle Integration 3</i>
SAP Adapter	<i>Using the SAP Adapter with Oracle Integration 3</i>
SAP Ariba Adapter	<i>Using the SAP Ariba Adapter with Oracle Integration 3</i>
SAP ASE (Sybase) Adapter	<i>Using the SAP ASE (Sybase) Adapter for Oracle Integration 3</i>
SAP Commerce Cloud (Hybris) Adapter	<i>Using the SAP Commerce Cloud (Hybris) Adapter with Oracle Integration 3</i>

Adapter	For Information
SAP Concur Adapter	<i>Using the SAP Concur Adapter with Oracle Integration 3</i>
SAP SuccessFactors Adapter	<i>Using the SAP SuccessFactors Adapter with Oracle Integration 3</i>
SAP S/4HANA Cloud Adapter	<i>Using the S/4HANA Adapter with Oracle Integration 3</i>
ServiceNow Adapter	<i>Using the ServiceNow Adapter with Oracle Integration 3</i>
Shopify Adapter	<i>Using the Shopify Adapter with Oracle Integration 3</i>
Shopify GraphQL Adapter	<i>Using the Shopify GraphQL Adapter with Oracle Integration 3</i>
Slack Adapter	<i>Using the Slack Adapter with Oracle Integration 3</i>
SOAP Adapter	<i>Using the SOAP Adapter with Oracle Integration 3</i>
Snowflake Adapter	<i>Using the Snowflake Adapter with Oracle Integration 3</i>
SugarCRM Adapter	<i>Using the SugarCRM Adapter with Oracle Integration 3</i>
SurveyMonkey Adapter	<i>Using the SurveyMonkey Adapter with Oracle Integration 3</i>
Trello Adapter	<i>Using the Trello Adapter with Oracle Integration 3</i>
Twilio Adapter	<i>Using the Twilio Adapter with Oracle Integration 3</i>
Twitter Adapter	<i>Using the Twitter Adapter with Oracle Integration 3</i>
Workday Adapter	<i>Using the Workday Adapter with Oracle Integration 3</i>
Zendesk Adapter	<i>Using the Zendesk Adapter with Oracle Integration 3</i>
Zuora Adapter	<i>Using the Zuora Adapter with Oracle Integration 3</i>

About Mappings

Most integrations include one or more mappings, which specify the source and destination of the data that the integration moves.

A mapping ensures that the appropriate data is taken from a source application and then placed in the correct location in the target application.

Example

Imagine you need to create a Jira account for a new employee. The source of the employee's personal information is Oracle HCM Cloud.

To create the account, an integration must move the employee's personal information from Oracle HCM Cloud (the source application) to Jira (the target application). The mapping for the integration identifies the following information:

- The database fields that contain the source data in Oracle HCM Cloud.
- The database fields that receive the data in Jira.

Mappings and connections work together in your integration. Connections identify the applications that an integration connects to, and a mapping identifies the data to move, including its source and destination fields.

When Terms or Values Differ, Use a Lookup

When you create a mapping, you choose the source and destination of the data that the integration moves. But what if the applications use different terms or values to describe the same thing? Use a lookup to translate the values so that the target application receives the expected value. See [About Lookups](#).

Connections Determine the Fields in the Trees

The connections determine the data structures that you see. For instance, a connection that is based on the Jira Adapter shows only the Jira database fields that are available.

When you create mappings, expand each tree as needed to find the correct field.

Deeper Dive into Mappings

The mappings that you create in the mapper are called transformation maps. These mappings use the eXtensible Stylesheet Language (XSL) to describe the data mappings, which let you perform complex data manipulation and transformation. A standard set of XSLT constructs are provided (for example, `xsl:if`, `xsl:for-each`, and others). A specialized function is also provided for you to reference lookups directly from the mapper.

 **Note:**

The mapper supports XSL version 2.0. Version 1.0 is not supported.

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.

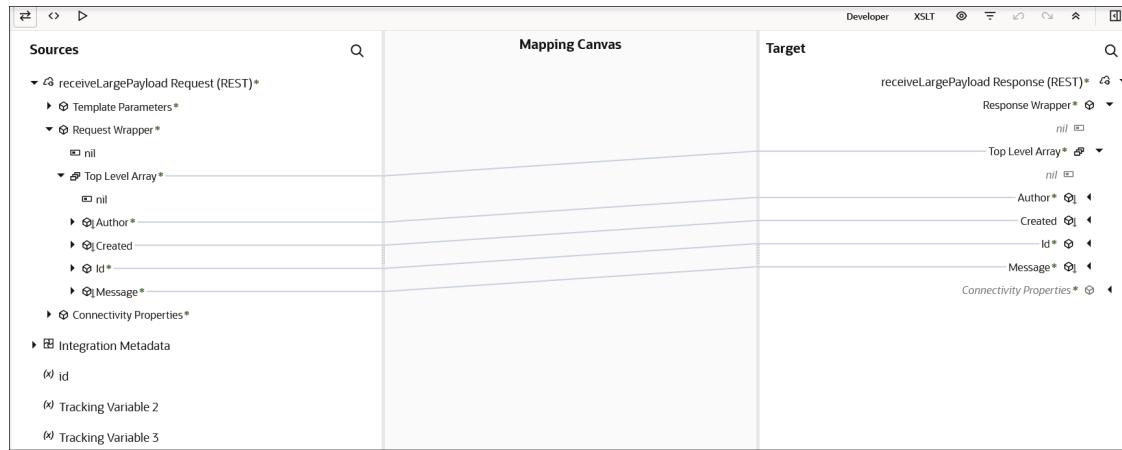
Substitution groups in schemas are supported. You can see all the substitutable elements in a base element in the mapper, and select the one to use.

Extended data types are also supported.

Use the Mapper to Define Mappings

After you add a connection to an integration, an empty mapping is created in the integration canvas.

To define the mapping in the mapper, simply double-click the mapping.



The mapper contains trees on the left and right and a canvas in the middle:

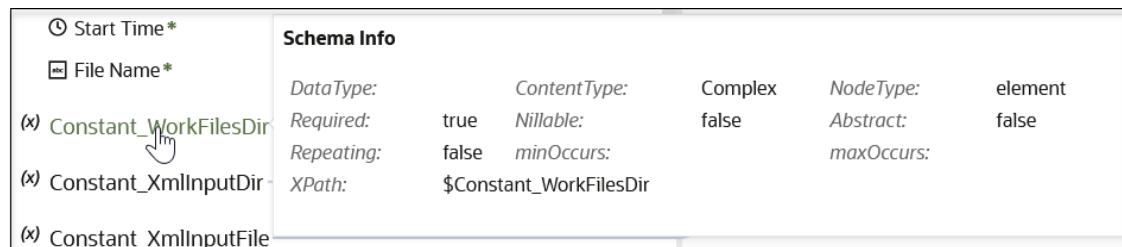
- The left tree is the data structure for the source application.
- The right tree is the data structure for the target application.
- On the canvas in the middle, view the lines that connect the entries in each tree.
An icon appears on a line when you update the mapping, such as by adding a function to it.

Connecting fields in the left and right trees is a simple drag-and-drop operation.

Tour of the Mapper

Elements and attributes for which mapping is required are identified by a blue asterisk (*) to the left of their names. To display only required fields, click the **Filter** icon in the mapper toolbar, select **All** from the **Fields** list and **Required** from the **Types** list, and click **Apply**.

You can also right-click elements and attributes and select **Node Info** to show specific schema details such as the data type, whether mapping is required, and so on.



Additional custom annotations can also be displayed. These annotations are currently only available with the Oracle CX Sales and B2B Service Adapter. The Oracle CX Sales and B2B Service Adapter obtains this information from the applications and annotates it in the integration WSDL. This information is then read and made visible as annotations in the mapper (for example, title and description). This information can help you better understand what data is being mapped.

Elements of the Mapper Toolbar

The mapper toolbar provides the following functionality.

Element	Description
	Click to return to the mapping canvas when you are inside the Code or Test page.
	View the XSLT code being created when you design your mappings.
	Once you complete designing your mappings, you can test them by entering sample content of the message to process in the mapping tester.
Developer	Click to disable user-friendly, source and target element names in the mapper. By default, user-friendly element names are shown.
XSLT	Click to show the XSLT functions.
	Select the following options: <ul style="list-style-type: none">• Show the namespace prefixes on source and target element nodes.• Show the types (prefixes and data types) on source and target element nodes.
	Filter the display of element nodes, error messages, and warnings in the source or target data structures.
	Select to undo the previous action performed in the mapper. For example, if you perform a mapping, then press this button, the mapping is removed. The link is disabled when all actions have been undone.
	Redo the action that was undone.
	Maximize the size of the mapper. This is useful when working with large schemas.
	Add functions, operators, and XSLT expressions to your mappings.

How to Create a Mapping

A separate guide walks you through creating mappings using the mapper. See Get Started with the Mapper in *Using the Oracle Mapper with Oracle Integration 3*.

About Lookups

When you create a mapping, you choose the source and destination of the data that the integration moves. But what if the applications use different terms or values to describe the same thing? Use a lookup to translate the values so that the target application receives the expected value.

When Terms or Values Differ, Create a Lookup

A lookup associates the value that an application uses for an item with the value used by other applications for the same item. A lookup is a table, and it can define one or more terms or values for two or more applications.

An integration might require you to include several fields in a lookup, depending on the applications that you're integrating and the number of fields that require mapping.

Examples

Consider an integration in which the source application uses numeric country codes and the target application uses alphabetical country codes. In such a case, add all the country code values to a lookup so that the mapping translates the numeric country codes to alphabetic country codes and displays the expected value in the target application.

Other examples of fields that might require lookups include currency codes and gender codes.

What Happens When You Don't Create a Lookup

If the applications that you're integrating use different terms or values and you don't create a lookup, the target application won't receive data the way that you expect. For example, you might attempt to copy a numeric country code into an alphabetic country code field.

Available Either Globally or Within a Project

A lookup is automatically available in multiple integrations, with some limitations:

- When you create lookups in a project
 - The lookups are available only within the project. You can use the lookups in any mapping that you create in any integration in the project.
- When you create lookups outside a project
 - The lookups are available only outside projects. You can use the lookups in any mapping that you create in any integration, as long as the integration is not in a project.

A Lookup Ensures That the Destination Application Receives the Correct Value

When an integration that calls a lookup runs, the integration checks the values in the lookup and matches the different codes. For example, if a source application provides a country code of 91 for India, the lookup table matches the value to alphabetic code of IN and sends the IN value to the destination application.

Lookup values are static. Runtime activities reference them but don't change them.

See [Map Data and Create Lookups](#).

About Packages

If you upgraded from Oracle Integration Generation 2, you might have some integrations in packages. In general, Oracle recommends designing integrations in projects rather than packages.

Packages Group Integrations

A package is a structure in which you can group one or more integrations. Packages let you easily import and export a group of integrations in Oracle Integration.

The Oracle Cloud Marketplace offers packages with prebuilt integrations that Oracle provides. You can import these packages to save time designing your integrations. You can also import and export packages that contain integrations that you or others have created.

Use Projects, Not Packages

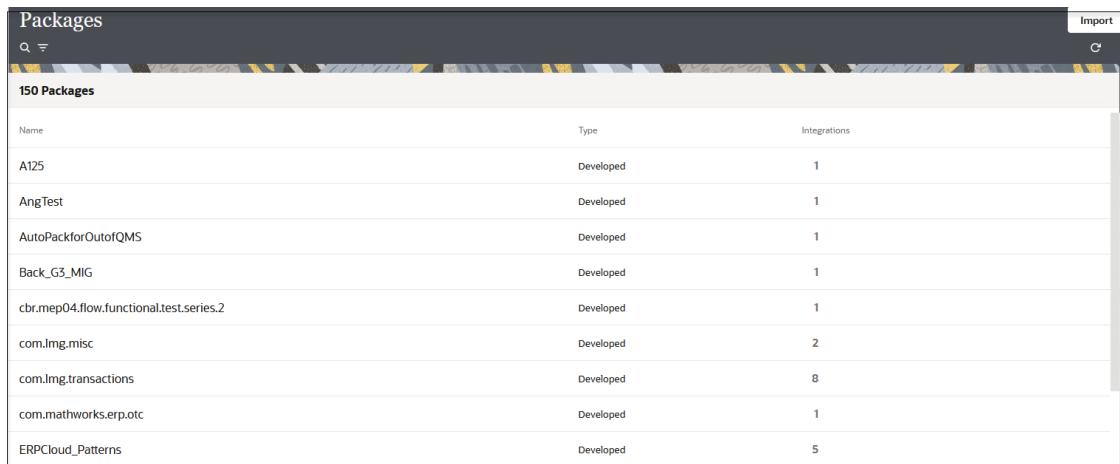
Packages are still available to use; however, Oracle recommends using projects instead. See [Get Started with Projects](#).

If you decide to use packages to group your integrations, remember that every package must include at least one integration. You can't lock a package to exclude other users of your Oracle Integration instance.

Packages Page

You create a package in the Create integration dialog, which appears when you create an integration. You can also update an integration's package in the Primary Information dialog.

View, delete, import, and export packages on the Packages page.



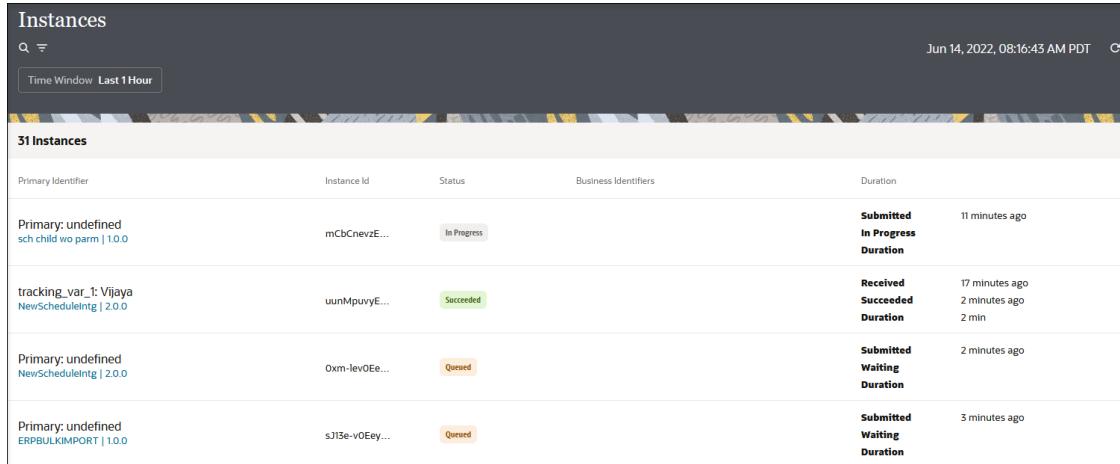
The screenshot shows a table titled "Packages" with a header row containing "Name", "Type", and "Integrations". Below the header, there are 150 rows of package data. The first few rows include "A125" (Developed, 1 integration), "AngTest" (Developed, 1 integration), "AutoPackforOutofQMS" (Developed, 1 integration), "Back_G3_MIG" (Developed, 1 integration), "cbr.mep04.flow.functional.test.series.2" (Developed, 1 integration), "com.lmg.misc" (Developed, 2 integrations), "com.lmg.transactions" (Developed, 8 integrations), "com.mathworks.erp.otc" (Developed, 1 integration), and "ERPCloud_Patterns" (Developed, 5 integrations). An "Import" button is located in the top right corner of the table area.

See [Manage Packages](#) and [Create an Integration](#).

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 Instances page displays information about the status of business identifiers and their values in your integrations.



The screenshot shows a table titled "Instances" with a header row containing "Primary Identifier", "Instance Id", "Status", "Business Identifiers", and "Duration". Below the header, there are 31 rows of instance data. The first few rows include "Primary: undefined sch child wo perm | 1.0.0" (mCbCnevzE..., In Progress, Submitted In Progress Duration 11 minutes ago), "tracking_var_1: Vijaya NewScheduleIntg | 2.0.0" (uumMpunyE..., Succeeded, Received Succeeded Duration 17 minutes ago 2 minutes ago 2 min), "Primary: undefined NewScheduleIntg | 2.0.0" (0xm-lev0Ee..., Queued, Submitted Waiting Duration 2 minutes ago), and "Primary: undefined ERPBULKIMPORT | 1.0.0" (sJ13e-v0Eey..., Queued, Submitted Waiting Duration 3 minutes ago). A timestamp "Jun 14, 2022, 08:16:43 AM PDT" and a "Time Window Last 1 Hour" filter are visible at the top of the table.

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

See [Assign Business Identifiers for Tracking Fields in Messages](#), [Track Integration Instances](#), and [Manage Errors](#).

About Error Management

You can manage integration errors from the Errors page in Oracle Integration. The Errors page displays information about individual integration instances and groups 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 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 activity stream and message payload of a failed integration instance
- View the business identifiers of a failed integration instance

Integration Failure Scenarios

Integrations can fail for reasons such as the following:

- 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.
- Any other Oracle Integration system failure.

See [Manage Errors](#).

Get Started with Observability

No matter your role or the number of integrations and robots that you're responsible for monitoring, you can find the information you need to keep your business on track in Oracle Integration.

Key Observability Features

Several sets of monitoring pages and customizable email messages provide you with the critical insights that you need to determine whether your business processes are running efficiently and error-free.

1. Observability pages in Oracle Integration

To monitor all your integrations in one place, head to the Observability pages, which you open from the navigation pane. The pages provide information about all the components in your Oracle Integration instance, including integrations that are and aren't in projects.

See [About the Observability Pages](#).

2. Observe page in a project

To monitor all automation solutions in a project, head to the Observe page in the project.

See [About the Observability Pages](#).

3. Daily or hourly notification emails

Quickly identify whether integrations have experienced errors by configuring notification emails. The emails provide reports about message counts, including failed messages, as well as alerts about other failures, including when a connectivity agent becomes unavailable.

See [Send System Status Reports with Notification Emails](#).

Develop an Observability Strategy

An observability strategy helps you plan your monitoring work for the business processes that your organization has automated. Whether simple or complex, an observability strategy helps you focus your time and energy on the tasks that matter most.

See [Develop an Observability Strategy](#).

Explore the Observability Tasks

Ready to learn about all the monitoring tasks in Oracle Integration? See [Monitor Integrations During Runtime](#).

Extend the Observability Pages to a Custom Application

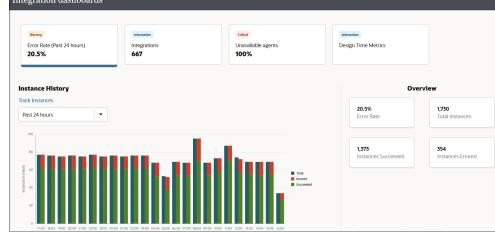
Want to see the observability information in your own monitoring tools? Use our REST APIs to access these capabilities. See [About Building Your Own Observability Tools](#).

About the Observability Pages

Oracle Integration includes several sets of pages for monitoring integrations and their components, including the Observability pages, which you open from the navigation pane, and the Observe page within a project.

About the Observability Pages

The Observability pages, which you open from the navigation pane, provide information about all the components in your instance, including integrations that are and aren't in projects.

Page	Description	What you can do on the page
Dashboards	When you want to check the overall health of everything your organization has built, head to the interactive Dashboards page, which provides at-a-glance information about the messages that Oracle Integration processed and the integrations that ran.	<ul style="list-style-type: none"> Understand the overall status and tracing levels for integrations. See the number of active integrations you have and the patterns that they use. Identify the connections that have the most errors. Determine whether your connectivity agents are up and running as expected. Understand the current status of the components your organization is developing, including projects, packages, integrations, connections, and more.
		
		<p>See View the Dashboard.</p>
Design time audit	Who updated an integration or other component, and when did the update happen? Get this information on the Design time audit page.	<ul style="list-style-type: none"> Review the actions that people took, the components they acted upon, the people who took the actions, and the dates and times when the actions occurred. <p>See Check the Audit History for an Integration or Other Component.</p>
Integrations	The Integrations page is where to go when you want to see high-level information about your integrations and their messages.	<ul style="list-style-type: none"> View the processing status for the messages that each integration generated, including the total number of success and error messages. Check performance over time of an integration, including viewing the fastest, slowest, and average run times for an integration, as well as the largest deviation from the average. Drill down into an integration, such as determining why an integration was aborted. <p>See Monitor the Message Processing Status of Integrations.</p>

Page	Description	What you can do on the page
Subscriptions	If you have integrations that use the Event pattern, open the Subscriptions page to get status information about these integrations.	<ul style="list-style-type: none"> Monitor the status of integrations that subscribe to events. Retain an event subscription when deactivating an integration.
		See Monitor Integrations That Subscribe to Events .
Agents	When the Dashboards page tells you that one or more connectivity agents is unavailable, open the Agents page to identify the agent that is down.	<ul style="list-style-type: none"> Identify the connectivity agents that are currently unavailable.
		See Monitor Agents .
Instances	When an integration runs, Oracle Integration creates an integration instance for it. Track the status of all your integration instances and view their message payloads on the Instances page.	<ul style="list-style-type: none"> Collect information about the type and use of each integration. Check the status of an individual integration instance. View the tracing level and the activity stream for an integration instance, see how the message flow through the integration went, and download the logs for the activity stream. Determine where an error is occurring in an integration. Abort an integration instance. Resubmit a failed integration instance.
		See Track Integration Instances .
Errors	Like the Instances page, the Errors page lists integration instances. However, while the Instances page lists <i>all</i> integration instances, the Errors page lists only the integration instances that <i>failed</i> .	<ul style="list-style-type: none"> Many tasks from the Instances page, plus: Quickly identify any failed integration instances. Resubmit or abort a failed or in-progress integration instance. Resubmit or abort multiple integration instances at one time. View only recoverable or nonrecoverable integration instances. View only the errored integration instances for a specific integration or connection, or for an integration that you have attempted to recover by resubmitting the integrations.
		See Manage Errors .
Future runs	On the Future runs page, view all the schedule integrations and the times they're scheduled to run. This page helps you understand the timeline of your integrations and the queue so that you don't schedule too many integrations to run at the same time.	<ul style="list-style-type: none"> Understand the schedule integrations in the queue. Determine whether your schedule integrations are clustered too closely.
		See View the Calendar of Schedule Integration Runs .

Page	Description	What you can do on the page
B2B tracking	If you use B2B for Oracle Integration, use the B2B tracking page to track information related to your B2B integrations.	<ul style="list-style-type: none"> Track the B2B message interaction between trading partners, including viewing wire messages. Wire messages are raw messages that trading partners exchange. Resubmit outbound messages. View message logs, which are similar to the activity stream for an integration and which show messages from trading partners, the identity of the trading partners, the unpacking of the messages, the status of the acknowledgement messages, and more. <p>See Track B2B Messages in Using B2B for Oracle Integration 3.</p>
Fusion Applications	When an integration that consumes a published event from an Oracle Fusion Application fails, troubleshooting the failure can be difficult. In such cases, head to the Fusion Applications page, where you can determine the source of the error.	<p>This page is only for integrations that use the Oracle ERP Cloud Adapter or the Oracle CX Sales and B2B Service Adapter (previously known as the Oracle Engagement Cloud Adapter).</p> <ul style="list-style-type: none"> When an event doesn't arrive as expected, determine whether the failure occurred in Oracle Fusion Applications or in Oracle Integration. Review and manage issues related to business event integrations. <p>See Diagnose and Manage Event-Based Oracle Fusion Applications Integrations.</p>

About the Observe Page in a Project

On the Observe page in a project, you can monitor the automation solutions in the project. Many tabs are identical to their counterpart pages in the Observability area, with one key difference: The Observe pages show information only for the automation solutions that are *in the project*.

See [Monitor Integrations in a Project](#).

Develop an Observability Strategy

If you're responsible for monitoring automation solutions, consider developing an observability strategy so that you can meet your business objectives while streamlining your monitoring work.

On This Page

- [Why You Need an Observability Strategy](#)
- [Tasks to Include](#)
- [Factors to Consider](#)
- [People Are Just as Important as Tasks](#)
- [Sample Observability Strategy](#)

Why You Need an Observability Strategy

Your options for observability are numerous. But how do you plan your monitoring work so that you find issues as early as possible while still allowing you time to complete your other responsibilities? And how do you ensure that everyone in your organization is in agreement about your approach to monitoring?

The answer is to create an observability strategy. An observability ensures that your organization discovers any issues with its business processes in a timely manner.

When creating an observability strategy, consider your work, including:

- The tasks that you perform daily, weekly, monthly, and quarterly.
- The time when you perform the tasks.
- The person who performs the task.
- The reason you perform the tasks.
- The stakeholders to notify in the event of an issue, and when they should be notified.
- The steps you take to determine the source of an issue.
- The steps you take to resolve an issue.

Tasks to Include

Monitoring tasks typically fall into one of the following categories:

- Tasks you complete before an integration goes live.
- Daily monitoring tasks to make sure everything is running as expected.
- Occasional tasks to ensure operational excellence.

For a collection of suggested tasks to potentially include in your observability strategy, see [Workflow for Monitoring Integrations](#).

Factors to Consider

Consider an organization with business-critical integrations that run daily at 10 AM and 4 PM. You might choose to actively monitor these integrations at those times. For other integrations, you might rely upon an hourly notification email or periodic reviews of the Observability pages to inform you of issues.

When planning your observability strategy, consider your business requirements, including:

- Which integrations are critical to my organization?
- How soon do we need to know about issues that impact our critical integrations? What about less critical integrations?
- What are the business impacts if we don't addresss the issues in a timely manner?
- How often do we need to check in to make sure everything is running as expected?

People Are Just as Important as Tasks

Don't forget the critical step of getting stakeholders involved early in the planning of the observability strategy. Getting all stakeholders to contribute to and agree to an observability strategy is sometimes a challenging task. However, consensus on the strategy contributes to a stronger strategy that is likely to meet your organization's monitoring requirements.

Sample Observability Strategy

An observability strategy doesn't need to be lengthy to be effective. For example, consider the following simple plan as a starting place for your organization.

Time	Task	Owner	Details
9:00 AM	Start-of-day status check	Arya Backups: Melissa, Mark	Review the hourly notification emails that came in overnight or the Observability pages and determine the following information: <ul style="list-style-type: none"> Did any integrations experienced errors overnight? Are all connectivity agents are up and running?
10:00 AM	Business-critical monitoring Monitor the business-critical Order Processing integration	Rajan Backups: Arya, Sneha	This integration runs twice every weekday. Review the hourly notification email or the Observability pages and determine the following information: <ul style="list-style-type: none"> Is the integration running without issues? Does the performance of the integration meet your organization's requirements? Are all connectivity agents are up and running?
11:00 AM	Hourly connectivity agent check	Arya Backups: Melissa, Mark	Determine whether all connectivity agents are up and running.
12:00 PM	Hourly connectivity agent check	Arya Backups: Melissa, Mark	Determine whether all connectivity agents are up and running.
1:00 PM	Midday status check and integration count check	Arya Backups: Melissa, Mark	Review the hourly notification email or the Observability pages and determine the following information: <ul style="list-style-type: none"> Are all integrations running without issues? Are all connectivity agents are up and running? Have the expected 120 integration instances run so far today?
2:00 PM	Hourly connectivity agent check	Arya Backups: Melissa, Mark	Determine whether all connectivity agents are up and running.
3:00 PM	Hourly connectivity agent check	Arya Backups: Melissa, Mark	Determine whether all connectivity agents are up and running.
4:00 PM	Business-critical monitoring Monitor the business-critical Order Processing integration	Rajan Backups: Arya, Sneha	This integration runs twice every weekday. Review the hourly notification email or the Observability pages and determine the following information: <ul style="list-style-type: none"> Is the integration running without issues? Does the performance of the integration meet your organization's requirements? Are all connectivity agents are up and running?
5:00 PM	End-of-day status check and integration count check	Arya Backups: Melissa, Mark	Review the hourly notification email or the Observability pages and determine the following information: <ul style="list-style-type: none"> Are all integrations running without issues? Are all connectivity agents are up and running? Have the expected 260 integration instances run so far today?

About Building Your Own Observability Tools

The observability tools in Oracle Integration provide robust capabilities for monitoring your business. You can use Oracle Integration's REST APIs to provide the same monitoring capabilities in your own observability tools.

When to Build Your Own

The monitoring capabilities offer insights into only the automation solutions that you create in Oracle Integration. However, your organization might prefer a monitoring tool that tracks work across multiple business applications.

In such cases, be aware that all of the monitoring capabilities that are available in the user interface are also available from the Oracle Integration REST APIs. If needed, you can create custom monitoring tools, such as custom user interfaces or scripts that run reports.

Review the Monitoring REST APIs

See [Monitoring REST Endpoints](#) in *REST API for Oracle Integration 3*.

About Oracle Integration

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

Oracle Integration provides native connectivity to Oracle Software as a Service (SaaS) applications, such as Oracle CX Sales and B2B Service Adapter, Oracle Fusion Cloud B2C Service, and so on. Oracle Integration *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 *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.

Start Oracle Integration

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

You receive your user account when you subscribe to Oracle Integration.

To access Oracle Integration:

1. In a web browser, enter the URL of your instance that you received from your administrator.
2. On the login page, enter your user name and password.

Once you sign in, the Home page is displayed. Select any item on this page to learn more about Oracle Integration and its features.

3. Click **Home**.

See Get Familiar with the Home Page in *Getting Started with Oracle Integration 3*.

Keyboard Shortcuts in the Integration Canvas

Use keyboard shortcuts to navigate and select options in the integration canvas in Oracle Integration.

 **Note:**

These keyboard shortcuts are available only in Oracle Integration 3. They are not supported in Oracle Integration Generation 2.

Table 1-1 General Keyboard Shortcuts

Task	Keyboard Shortcut
Open keyboard shortcuts reference	Shift + ?
Focus canvas	Shift + C
Change focus between tool bars	Shift + T
Navigate toolbar	Tab or  

Table 1-2 Canvas Keyboard Shortcuts

Task	Keyboard Shortcut
Save integration	Shift + S
Open integrations page	Shift + B
Zoom in	+
Zoom out	-
Scroll	Alt +     fn +     Home Page Up Page Down End
Navigate: Focus action to the left	
Navigate: Focus action to the right	
Navigate: Focus action above	

Table 1-2 (Cont.) Canvas Keyboard Shortcuts

Task	Keyboard Shortcut
Navigate: Focus action after	
Go to: First action	1
Go to: Last action	2
Go to: Selected action	3
Design: Select or deselect action	Enter
Design: Open action menu	M
Design: Insert action before	B
Design: Insert action after	A
Design: Insert action into	I
Design: Cut	Ctrl + X
Design: Paste	Ctrl + V
Design: Cancel cut and paste	Esc

Best Practices

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, and much more. Click one of the following topics to learn more about integration patterns, best practices for designing an integration correctly from the beginning, trigger (inbound) and invoke (outbound) connections to your applications, and integration creation steps.

Topics:

- [Avoid Common Integration Pattern Pitfalls](#)
- [Design Best Practices](#)

Avoid Common Integration Pattern Pitfalls

Designing integrations correctly from the start can save you a tremendous amount of rework. This section describes common integration pattern pitfalls (known as antipatterns) and best practices for avoiding these pitfalls.

- [Chatty Integrations](#)
- [Scheduled Job that Never Stops Trying to Process](#)
- [Import an Externally Updated IAR File](#)
- [Synchronous Integration Doing Too Much](#)
- [Too Many Connections in an Integration](#)
- [Read Files with Many Records](#)
- [Integrations Running Unchanged Despite Changing Business Needs](#)

Chatty Integrations

Use Case: Synchronize records in a file or large data set with an external system (for example, synchronize journal transactions or upload employee records into Oracle HCM Cloud).

Antipattern	Why an Antipattern?	Best Practice
Use an invoke activity within a looping construct to call external APIs for every record.	<ul style="list-style-type: none"> Downstream applications are receiving a large number of atomic requests. This puts the entire system under duress. A usage-based pricing model translates to high costs. 	<ul style="list-style-type: none"> Leverage application capabilities to accept multiple records in a single request: <ul style="list-style-type: none"> Salesforce: 200 records, Oracle Engagement Cloud/Oracle ERP Cloud: 100 records, Oracle Service Cloud: 1000 records Leverage adapter capabilities to send a large data set as attachments/files. <ul style="list-style-type: none"> Salesforce Adapter: 10,000 records/10 MB files, Oracle ERP Cloud Adapter supports FBDI files. Use a stage file action (the append to file option) to send the file to the destination at the end.

Scheduled Job that Never Stops Trying to Process

Use Case: Process records within a set of files with a tight service level agreement (SLA) (for example, synchronize employee records into Active Directory from Oracle HCM Cloud or send benefits information).

Antipattern	Why an Antipattern?	Best Practice
The schedule integration looks for all files to process and loops over all to sequentially process until no files remain.	<ul style="list-style-type: none"> If a large number of files exist, one run of a scheduled job processes for a long time and starves other jobs and may get terminated by the framework. Processing tied to a single server: this does not leverage multiple nodes in a cluster. 	<ul style="list-style-type: none"> Limit the number of files to process in a single scheduled run. <ul style="list-style-type: none"> Limit results from the list file. Use schedule parameters to remember the last processed file for the next run. Invoke the run now command from the same integration to trigger processing of the next file if waiting for the next scheduled run is not feasible.

Import an Externally Updated IAR File

Use Case: Need to leverage advanced XSL constructs that may not be available in the mapper.

Antipattern	Why an Antipattern?	Best Practice
Update the IAR file externally and then import it into Oracle Integration.	<ul style="list-style-type: none"> This can lead to metadata inconsistency and validation failures. Activation failures may occur. 	<ul style="list-style-type: none"> Use the import map feature in Oracle Integration. <ul style="list-style-type: none"> Keeps metadata consistent and leverages validation. The mapper can show an imported map and allow it to be edited.

Synchronous Integration Doing Too Much

Use Case: A request triggers complex processing involving enrichment and updates across multiple systems.

Antipattern	Why an Antipattern?	Best Practice
<ul style="list-style-type: none"> Huge synchronous integrations designed with a large number of invokes / conditional logic. Synchronous integrations with invokes within a loop with a large number of iterations. 	<ul style="list-style-type: none"> Susceptible to timeouts. Any marginal slowdown adds up. Blocking call. Holds resources and starves other integrations. 	<ul style="list-style-type: none"> Consider moving completely to an asynchronous integration (fire and forget, asynchronous response). <ul style="list-style-type: none"> The integration platform provides an acknowledgment to clients upon receiving the message. The integration platform ensures guaranteed processing and also supports resubmission of failures. Split into a synchronous integration containing mandatory processing before sending out a response and triggering separate asynchronous fire and forget integrations for other processing logic. Optimize synchronous processing with a coarse-grained external API to replace multiple chatty calls.

Too Many Connections in an Integration

Use Case: When developers create integrations, they define their own connections pointing to the same application. This leads to many duplicate connections.

Antipattern	Why an Antipattern?	Best Practice
Every developer creates their own connection using a different set of configurations/credentials.	<ul style="list-style-type: none"> A high number of connections make manageability painful, especially when you need to update the endpoint, credentials, configuration, and so on. Impact analysis is complicated when there is an application upgrade or metadata/coordinate change. 	<ul style="list-style-type: none"> Have a custodian create needed connections and ensure that duplicate connections of the same types are not created. Build a best practice for naming conventions and maintaining a set of configurations. Use the Configuration Editor to edit/replace the connection in an integration.

Read Files with Many Records

Use Case: Read a file with a large number of records and process individual records.

Antipattern	Why an Antipattern?	Best Practice
Read the entire file in memory using the read file option and process it record by record.	<ul style="list-style-type: none"> Large amounts of memory are consumed and other system processing is impacted. Sequential processing does not leverage built-in map reduction capabilities. 	<ul style="list-style-type: none"> Download the file to the staged location using the download file option. Stage files are secured and deleted at the completion of the instance. Use the read file with segments option. <ul style="list-style-type: none"> You can still model record level processing inside the segment processing. The platform automatically processes segments in parallel. The platform brings in only portions of the file to memory, when needed.

Integrations Running Unchanged Despite Changing Business Needs

Use Case: Integrations/schedules created during the initial implementation continue to run even though your business requirements have changed over time.

Antipattern	Why an Antipattern?	Best Practice
<ul style="list-style-type: none"> Integrations and scheduled jobs created during the initial product implementation are never re-evaluated against changing business needs. A proliferation of integrations occurs without looking at existing integrations. 	<ul style="list-style-type: none"> Unnecessary runs of jobs that handle no work. Unoptimized invocations of multiple integrations with similar functionality. Clutter with dead integrations, lifecycle management (LCM) overhead, and developer confusion. 	<ul style="list-style-type: none"> Periodically analyze existing integrations/schedules against current business needs. <ul style="list-style-type: none"> Look at monitoring data for historical runs. Consolidate integrations that are very similar. <ul style="list-style-type: none"> Integrations A and B subscribing to the Customer Created event in Oracle Engagement Cloud: integration A for invoking endpoint A and integration B for invoking endpoint B. Deactivate nonrelevant integrations. <ul style="list-style-type: none"> An integration for generating a file meant for partner consumption becomes irrelevant when the partner no longer needs it. Adjust the schedules for integrations that need lesser frequency and delete schedules no longer required. <ul style="list-style-type: none"> Files generated monthly have schedules run every few minutes. Changes in business needs can make certain schedules completely unnecessary.

Design Best Practices

Understand the following best practices and integration pattern pitfalls to avoid before you design an integration.

- [Create Schedule Integrations Mindfully](#)
- [Follow Guidelines for Synchronous Integrations](#)
- [Design Long-Running Integrations as Asynchronous](#)
- [Understand Timeouts in Synchronous Integrations](#)
- [Send Data to Third-Party Systems Efficiently](#)

Create Schedule Integrations Mindfully

When there are too many schedule integrations configured, instances can become backlogged waiting for resources to become available or previous integration runs to complete. This can

cause processing delays where some instances are in a waiting state longer than they must be and schedules may not start at the scheduled time.

As a best practice, do not create too many schedule integrations that are scheduled to run at the same time. Where possible:

- Use an asynchronous REST Adapter trigger instead of a scheduled trigger when an active schedule is not absolutely required.
- Do not create any long-running schedule integrations (a schedule integration that takes longer than one hour, for example, to complete). This blocks scheduler resources impacting other scheduled runs.
- Spread schedules out over time to avoid schedule clusters.

You can convert a schedule integration to a REST Adapter trigger-based application integration. See [Convert a Schedule Integration to a REST Adapter-Triggered, Application Integration](#).

If you absolutely require a large number of schedule integrations, and you encounter the previously described issue, the following design changes as a solution are recommended:

1. For each schedule integration, convert it to a REST Adapter trigger-based application integration. See [Convert a Schedule Integration to a REST Adapter-Triggered, Application Integration](#).
2. Create a new schedule integration that only performs an asynchronous invoke of the application integration you converted in step 1 above.

This solution enables the schedule integration to start at the scheduled time, invoke the REST Adapter trigger-based child integration asynchronously, and complete within milliseconds. This approach reduces the backlog and contention surrounding scheduler resources.

If you have a large number of schedule integrations to convert, a staged approach starting with the following integrations is recommended.

- Longest-running schedule integrations.
- Schedule integrations configured with the shortest frequency (for example, integrations that run every 10 minutes or less).

Design any new schedule integrations with the design practices described above.

See [Schedule Integrations Are Not Running on Time](#) and [When a Schedule Integration Instance Gets Terminated](#).

Follow Guidelines for Synchronous Integrations

Note the following best practices when designing a synchronous integration.

- A synchronous integration calling any asynchronous request response service:
 - Calling asynchronous fire and forget (one-way) is acceptable.
 - Oracle Integration does not currently allow modeling an asynchronous request response service. However, all schedule integrations internally use an asynchronous request response. Therefore, a synchronous integration using a schedule integration is an antipattern.
- A synchronous integration calling multiple services that span more than five minutes is reported as a stuck thread.

Design Long-Running Integrations as Asynchronous

Note the following best practice when designing long-running or time-consuming integrations.

Do not expose long-running or time-consuming integrations as synchronous flows. This action can lead to client applications (including other integrations) timing out. Synchronous integrations also have a server-side timeout. Instead, model any synchronous integration taking more than two minutes as an asynchronous flow.

Understand Timeouts in Synchronous Integrations

You may have scenarios in which synchronous invocations from Oracle Integration (including calls to other integrations) are blocking calls and must complete within 300 seconds.

Because the call may involve one or more proxies, each of the proxies may have a similar timeout. For instance, the default proxy on Oracle Public Cloud has a timeout value of 120 seconds. If the call is to an on-premises service is behind a fire wall, the proxy configured may also have its own timeout value.

In case timeouts are defined at multiple layers, the service invocation fails at the first timeout.

Send Data to Third-Party Systems Efficiently

While there is no specific integration design to automatically enable an outbound integration to send data to different third party systems in parallel, there are integration design approaches that allow this scenario, including using parallel processing in outbound integrations.

Separate the integration into multiple integrations:

- Create a main parent integration that only receives/processes the data.
- Create separate child integrations to perform the individual outbound REST invocations.

The interface between the main and separate child integrations can follow these approaches:

- Consist of dummy REST calls, but it must be asynchronous. Essentially, the asynchronous calls are not blocked by the response and the fire-and-forget design enables available threads to work on child integration processing in parallel, within the available system resources. This type of design is recommended because if all synchronous REST calls are done in the same integration, a timeout error may occur if the sum of time taken for each synchronous call exceeds five minutes.
- Follow a publish/subscribe design approach (for example, putting the data events in a queue, having each child flow subscribe from the queue, and so on).

Design and Monitor Integrations in Projects

You can design, manage, and monitor integrations from a single, self-contained workspace known as a project. This eliminates the need to navigate to other parts of the Oracle Integration user interface (which is required with standalone integrations or integrations in packages).

Topics:

- [Design a Project](#)
- [Manage a Project](#)
- [Monitor Integrations in a Project](#)

Before using integration projects, be familiar with the conceptual details. See [About Projects](#).

Design a Project

Create and design an automation solution, including all of its components, in a project.

Topics:

- [Control Who Can Edit, View, and Monitor in a Project](#)
- [Create or Import a Project](#)
- [Create, Import, or Copy an Integration into a Project](#)
- [Unlock an Integration in a Project](#)
- [Refresh Endpoints of an Integration in a Project](#)
- [Update Property Values of an Integration in a Project](#)
- [Create a Version of an Integration in a Project](#)
- [Add and Share a Connection Across a Project](#)
- [Edit a Connection in a Project](#)
- [Refresh Connection Metadata in a Project](#)
- [Delete a Connection in a Project](#)
- [Unlock a Connection in a Project](#)
- [Add a Lookup to a Project](#)
- [Edit a Lookup in a Project](#)
- [Clone a Lookup in a Project](#)
- [Delete a Lookup in a Project](#)
- [Import a JavaScript Library in a Project](#)
- [Delete a JavaScript Library in a Project](#)
- [Update a JavaScript Library in a Project](#)

- Create, Publish, and Subscribe to Events in a Project
- Create Robots in a Project
- Create Healthcare Integrations in a Project
- Create B2B Integrations in a Project

Control Who Can Edit, View, and Monitor in a Project

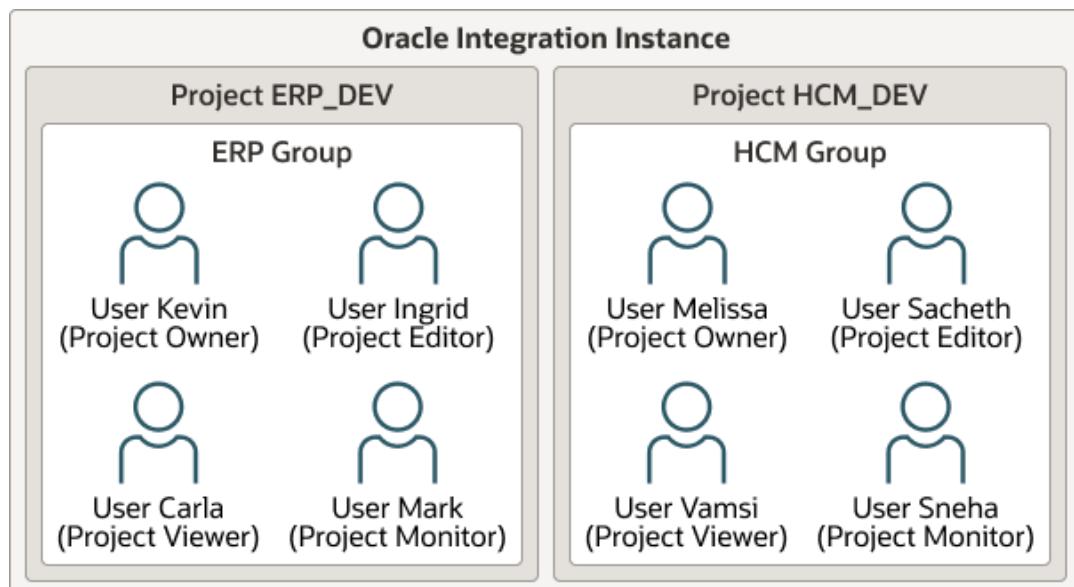
You can control the users and groups that edit, view, and monitor a project with role-based access control (RBAC). You select who can access a set of project resources while limiting (or hiding) those same resources from others (for example, providing an HCM group of users with access to some project resources while restricting an ERP group of users from accessing those same project resources).

- Capabilities
- RBAC-Enabled Projects FAQ
- Select Who Can Edit, View, and Monitor a Project
- Best Practices
- RBAC-Enabled Project Use Case

Capabilities

RBAC-enabled projects provide the following capabilities:

- Support project isolation
Project isolation enables multiple users and groups to work in the same Oracle Integration instance, with access to only the projects to which they are assigned. This enables multiple groups (for example, ERP and HCM) to work on the same Oracle Integration instance and only edit, view, and monitor project resources assigned to their group.



- Support for setting restriction levels within a project

You can assign users and groups to specific project permissions based on their needs. For example, you assign project editor permissions to some users and groups to create and

edit projects and invoke integrations, while you assign project monitor permissions to other users and groups to only monitor integration instances at runtime.



Project permissions are similar to the Oracle Integration service roles (for example, ServiceDeveloper, ServiceMonitor, ServiceInvoker, and others) that the user with the ServiceAdministrator role first assigns to other users and groups from the Oracle Cloud Infrastructure Console in an Oracle Identity Cloud Service or identity domain environment. See Manage Access and Assign Roles in *Provisioning and Administering Oracle Integration 3*.

Project permissions are essentially a layer of permissions on top of service roles.

Once you assign these service roles to users and groups, the user with the ServiceAdministrator role or the project owner can assign these users and groups with project permissions inside a specific project. Project access is restricted based on a user's service role, plus their project permissions.

The following project permissions are available:

PDescription

r
o
j
e
c
t
P
e
r
m
i
s
s
i
o
n

Enables you to perform all available actions on the project (as a project-level super user), including associating groups/users with each project permission. The project creator (owner) is automatically set as a project-level super user.

Modification of project permissions is restricted to the user with the ServiceAdministrator role (system super user) and the project owner (project-level super user).

Only a user with the ServiceAdministrator role or a ServiceDeveloper who is an owner/editor of a project can export the project to another Oracle Integration instance.

PDescription

r
o
j
e
c
t
P
e
r
m
i
s
s
i
o
n

EEEnables you to edit certain metadata in a project, create projects, and create, remove, edit, and run project resources.

i You can view runtime metrics and perform actions against runtime instances. You cannot modify the assigned project permissions.

o
r

VEnables you to view certain metadata in a project and discover and view project resources.

i You cannot add, remove, edit, run, export, or import a project or project resources.

e
w
e
r

MEnables you to monitor runtime metrics in a project and perform actions (for example, abort, retry, and discard) against runtime instances.

n You cannot access the **Design** and **Deploy** tabs of a project.

i
t
o
r

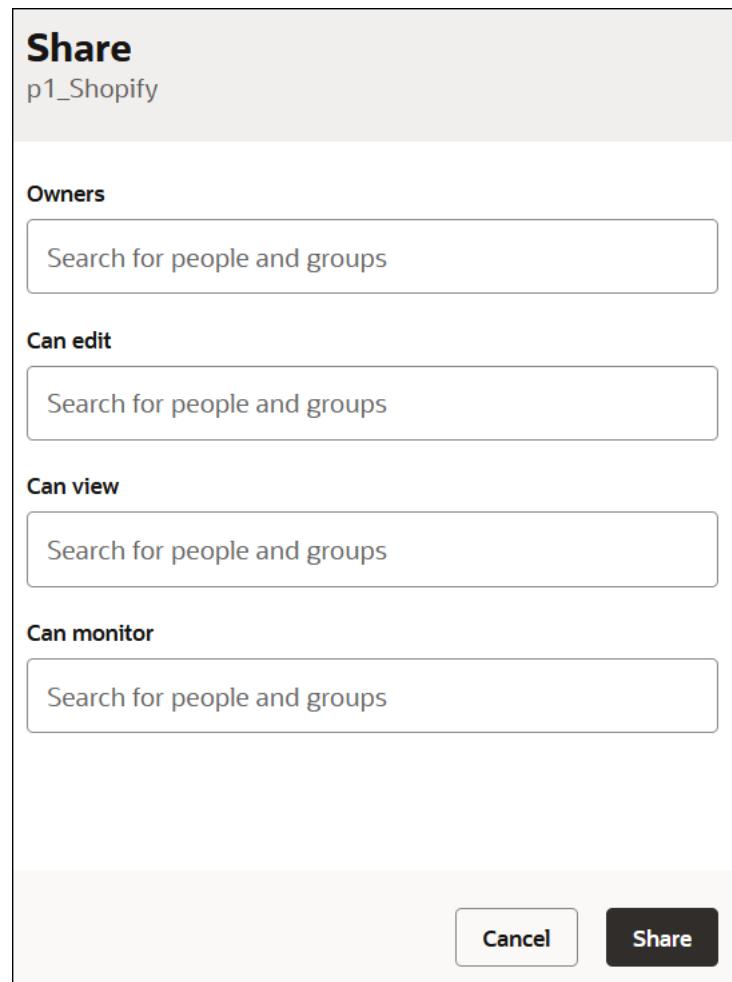
NA user *without* any project permissions (owner, editor, viewer, monitor) can see the project name when viewing the main Projects page. However, the user cannot do the following:

- n – Access the project details page to retrieve any details or query details with a REST API.
 - e – View instance monitoring details (for example, integration statistics, instances, errors, and more) outside the project under the **Observability** tab.
-

- Support RBAC for projects created prior to version 23.06

After version 23.06 patching, projects created in versions 23.04 and earlier automatically include the Share panel inside the project. These earlier projects automatically inherit the **Everyone** project permission. This setting makes everyone with the correct service role an owner who can edit, view, and monitor project resources. You can then restrict access if necessary by assigning users and groups to each project permission. If you are happy with your pre-23.06 permissions, you can also leave things as they are. Assigning specific project permissions is not required.

- Enforce the rule that a user's assigned service roles (ServiceDeveloper, ServiceMonitor, ServiceInvoker, and others) *always* takes precedence over their assigned project permissions. For example:
 - Assume a user with the ServiceMonitor service role is assigned the editor project permission in the **Can edit** field.



However, because this user is only assigned the ServiceMonitor service role, their permissions in the project are not elevated to allow them to create, update, or delete project resources; they can only monitor the project.

- Support projects on a single Oracle Integration instance that are both unrestricted and restricted for use by one or more groups.
When you create a project, import a project, or convert a package to a project, you can select the **Anyone can edit, view, and monitor** check box if you want to make a project unrestricted to all users with the correct service roles. See [Create or Import a Project](#) and [Convert a Package to a Project](#).
- Allow users with the ServiceAdministrator role to perform all actions in a project, regardless of project permissions.
- Filter the ability to call REST APIs.
For example, a user in an ERP group who invokes the `/ic/api/integration/v1/monitoring/instances` API only returns flow instance data from integrations attached to their group.
- Support having zero or more project owners.
If zero, then project owner defaults to the user with the ServiceAdministrator role. Only administrators have the unique privilege to update project-level access control. Project owners are essentially project-level super users.

RBAC-Enabled Projects FAQ

- **Can you import a standalone integration into an RBAC-enabled project?**
Only a user with the ServiceAdministrator role or a user with the ServiceDeveloper role that has the owner or editor project permission can import a standalone integration into an RBAC-enabled project.
- **Are project permissions moved from development to test to production environments in a CICD pipeline?**
Project permissions are not moved to test and production. The users and groups have different permissions in different environments. However, if a user sets up permissions in production, they are carried forward for future project updates. When a project is imported for the first time, it is owned by whoever imported it. That user must set the project permissions. If a project is re-imported, the permissions are left alone.
- **Do standalone (non-project) integrations use RBAC?**
Integrations created outside of a project do not support RBAC. Non-project or global resources are restricted by existing service roles.
- **Are there limits on the number of users and groups you can assign to a project?**
You can assign a maximum of five users and/or groups (any combination) to each of the project roles.
- **Do I need to use RBAC with my projects?**
No, using RBAC is optional. If you do not want to use it, you can ignore **Share**  in the project.
- **Can members of a project see the other members and their project permissions?**
Only a user with the ServiceAdministrator role or a ServiceDeveloper who is the project owner can see other assigned members. For these two conditions only, **Share**  is editable.
- **Do project permissions take precedence over service roles?**
No. Service roles (ServiceDeveloper, ServiceMonitor, ServiceInvoker, and others) always takes precedent over the assigned project permissions. For example, if a user with the ServiceMonitor service role is assigned the editor project permission, they cannot access the **Design** and **Deploy** tabs of a project.
- **Can restrictions be enforced at the REST API level? For example, can lookup update/delete only be assigned to a specific user and restricted from another?**
No. If both users have editor permissions on one type of resource, they have permissions on all types of resources.
- **If you have the editor permission in project A and want to invoke a child integration (set as publicly available) in project B, but you only have the monitor permission (or perhaps, no permission) in project B, can you do so? Or do you need to update your permissions in project B to match those in project A?**
You can discover and invoke the child integration in project B without setting any additional project permissions.
- **Any special permissions for project deployment? Can only a user with the edit permission create a deployment?**
To create a project deployment, you must have the ServiceAdministrator role or the ServiceDeveloper role plus the project owner/editor permission.
- **Can a user create groups and assign permission to groups rather than assigning users?**
Yes, the entries can be Oracle Identity Cloud Service or identity domain users or groups.
- **Can a user with the ServiceMonitor service role see all integrations?**

From monitoring pages under **Observability**, the user with the ServiceMonitor service role cannot see all the integrations. They cannot see integrations that are part of a project on which they do not have any permissions. The same applies to integration instances.

- **Are project roles applied in a production system?**

Project roles are still applicable. For example, the HCM_monitor group can monitor HCM projects, but not the finance project.

- **What can a user with no project permissions do?**

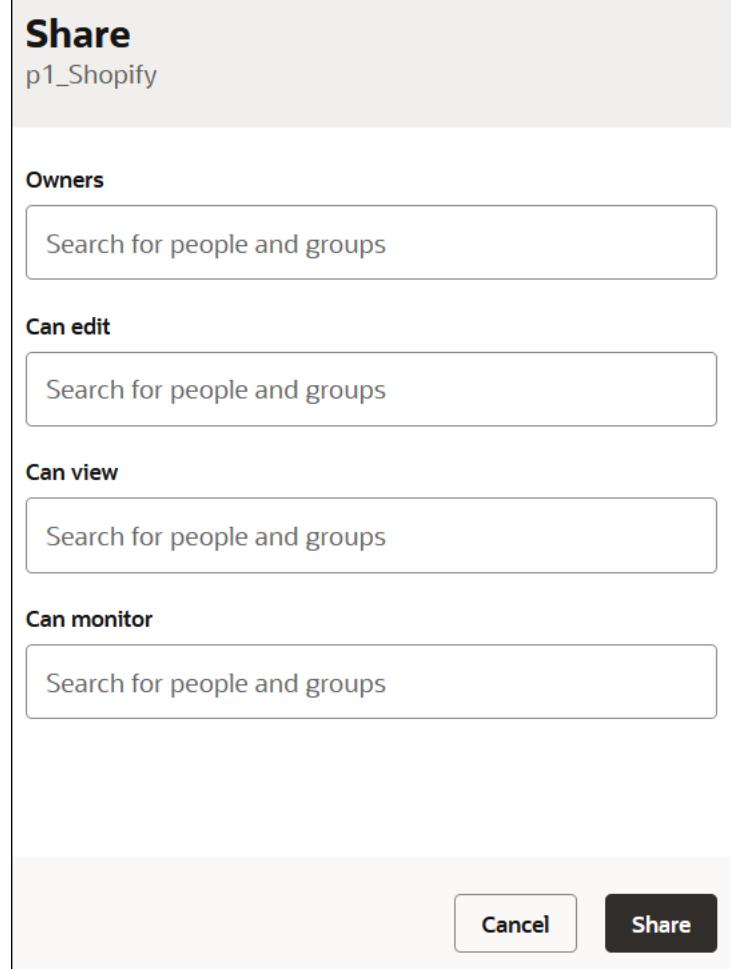
They can see the existence of the project on the main Projects page, but cannot perform any actions and are not allowed to access the project details page. They also cannot see monitoring resources (for example, integration statistics, instances, errors, and others) for the project under the **Observability** tab.

Select Who Can Edit, View, and Monitor a Project

The user with the ServiceAdministrator role or the project owner can select the users and groups who can view, edit, and monitor the page.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the upper right corner, click **Share** .

The Share panel is displayed.



- Assign users and groups to the project permissions, and click **Share** .

 **Note:**

You must have already created users and groups to which you assigned those users before they are visible for selection in this dialog. See *Manage Access and Assign Roles* in *Provisioning and Administering Oracle Integration 3*.

Field	Description
Owners	Begin typing to search for users and groups to own the project. By default, the user who created the project is an owner. The user with the ServiceAdministrator service role is also automatically an owner. The owner can perform all available actions, including associating users and groups with each of the project permissions.
Can edit	Begin typing to search for users and groups to which to assign the editor permission. If you select a user or group that includes users that do not have the ServiceDeveloper role, those users are unable to edit the project.
Can view	Begin typing to search for users and groups to which to assign the viewer permission.
Can monitor	Begin typing to search for users and groups to which to assign the monitor permission.

Best Practices

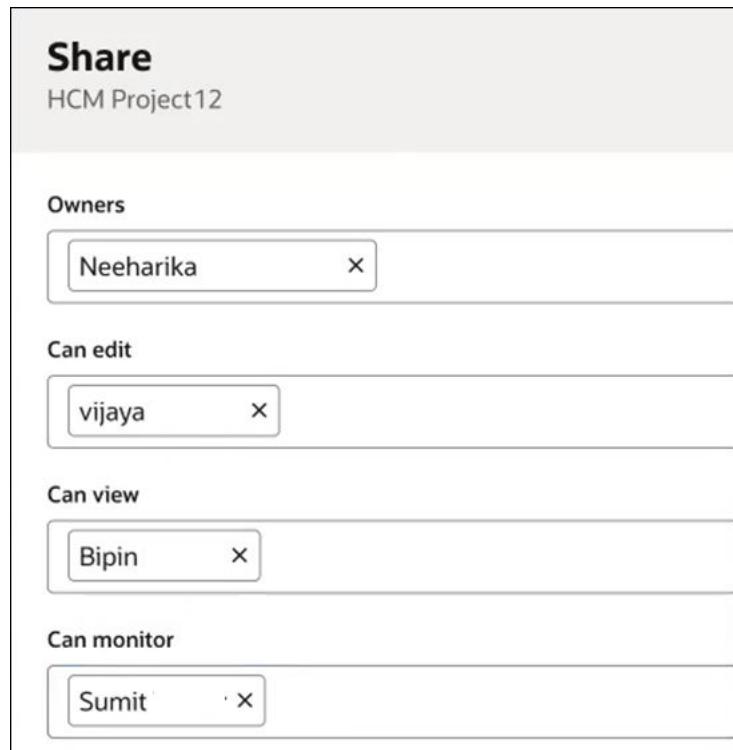
- Provide project access only to users that need it.
- Give users the minimal permissions necessary to perform their responsibilities. For example, if a user only needs to monitor integrations, don't give them the permission to edit integrations.
- Give projects nonsensitive names. For example, don't specify names such as employeeSalaries, employeeRatings, and so on. Even if users don't have access to projects, they can still see the project names listed on the Projects page.

RBAC-Enabled Project Use Case

This use cases provides a high level example of creating an RBAC-enabled project. It demonstrates what users can and cannot do based on their assigned project permissions.

- Neeharika, the user with the ServiceAdministrator role, performs the following tasks:
 - Creates the following user and service role assignments in the Oracle Cloud Infrastructure Console:
 - Vijaya: ServiceDeveloper
 - Bipin: ServiceDeveloper
 - Sumit: ServiceDeveloper

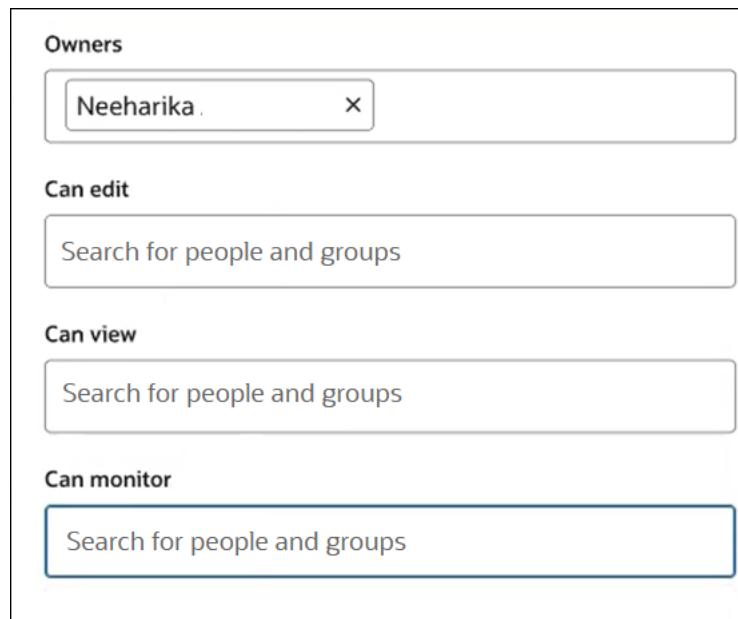
- b. Goes to the Projects page, and clicks **Add**.
- c. Creates two new projects:
 - **HCM Project12**
 - **FinancialServiceLocalInvoke**
- d. Opens **HCM Project12**.
- e. In the upper right corner, clicks **Share** .
- f. Assigns the following users to each project permission. By default, Neeharika is the owner because she created the **HCM Project12** project.



This configuration creates the following similarities between the service roles and the project permissions:

User	Service Role	Project Permission
Vijaya	ServiceDeveloper	Editor
Bipin	ServiceDeveloper	Viewer
Sumit	ServiceDeveloper	Monitor

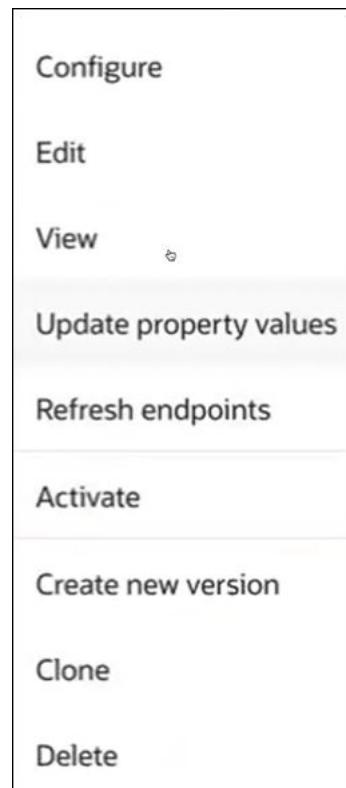
- g. Does not make any project permission updates under **Share**  of the **FinancialServiceLocalInvoke** project.
- Because Neeharika created this project, she is listed as the owner. The **Can edit** (for the editor permission), **Can view** (for the viewer permission), and **Can monitor** (for the monitor permission) fields remain empty.



With project permission assignments complete, Vijaya (editor permission), Bipin (viewer permission), and Sumit (monitor permission) log in and see their capabilities and restrictions.

2. What the Vijaya (ServiceDeveloper with the editor project permission) can do:
 - a. Opens the **HCM Project12** project and goes to **Share** . Because Vijaya has editor permissions, she can edit, view, and monitor resources in the project.

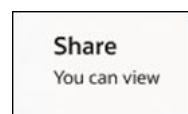
Share
You can edit, view and monitor
 - b. Clicks **Actions ...** in the **Integrations** section and notes that all edit actions are listed and can be performed.



- c. Moves around the project and successfully performs assorted editor tasks without any restrictions, such as the following:
 - Adds new project keywords in the **Details** section.
 - Imports a schedule integration and updates the schedule.
 - Deletes a library.
- d. Exits the **HCM Project12** project.
- e. Attempts to open the **FinancialServiceLocalInvoke** project. This is the project described in Step h to which no project permissions were assigned. Because Vijaya is not assigned any permissions on this project, the following error is displayed:

User Vijaya does not have sufficient privilege to perform this action.

3. What Bipin (ServiceDeveloper with the viewer project permission) can do:
 - a. Opens the **HCM Project12** project and, clicks **Share** . Because Bipin has viewer permissions, he can only view resources in the project.



- b. Clicks **Actions ...** in the **Integrations** section and notes that no edit actions are listed; only the **View** and **Schedule** (for viewing the schedule) actions can be performed.

Integrations			
→	Simple Schedule With Params	1.0.0	Configured
→	Fetch Emp Records	1.0.0	Configured

- c. Clicks the **Deploy** tab.
- d. Clicks **Actions ...** and notes that no edit actions are listed; only the **View** action can be performed.

Showing 2 deployment(s)			
Name	Identifier	Integrations	Last updated
2.0.0	02.00.0000	0	May 17, 2023 11:10:29 AM IST
1.0.0	01.00.0000	4	May 17, 2023 08:45 AM IST

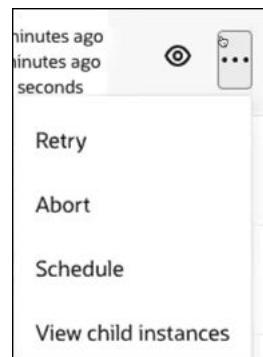
- e. Clicks the **Observe** tab.
- f. Clicks **Actions ...** and notes that no edit actions such as abort or retry are listed; only the **Schedule** (to view a schedule) and **View child instances** actions can be performed.

Primary: undefined Poll Emp Info 1.0.0	aXOijPTEe2F6m... Submitted Scheduled Duration	Queued	11 m in 4
startTime: 2023-05-17T15:06:33.6... Poll Emp Info 1.0.0	nAFD3vTBee2F6... Received Errored Duration	Errored	Schedule
start Time: 2023-05-17T14:50:15.5... Sync Emp Info 1.0.0	iZm-ifTCEe2K8b... Received Succeeded Duration	Succeeded	View child instances 22 minutes ago 5 min

4. What Sumit (ServiceDeveloper with the monitor project permission) can do:
 - a. Opens the **HCM Project12** project. Because Sumit has monitor permissions in the project, only the **Observe** tab is accessible. The **Design** and **Deploy** tabs are both disabled.

Design	Deploy	Observe
--------	--------	---------

- b. Clicks **Observe**.
- c. Clicks **View details** to view the activity stream.
- d. Click **Actions ...** and notes that **Retry**, **Abort**, **Schedule**, and **View child instances** can all be performed.



- e. Clicks the **Future runs** and **Audit** tabs to view future scheduled runs and audit details, respectively.



- f. In the navigation pane, clicks **Observability**, then **Dashboards** for a global view of aggregated data for all integrations in the instance (whether or not they are in a project). All integrations are visible to Sumit regardless of his project permissions.
- g. In the navigation pane, clicks the **Integrations** tab to view all integration instances except for instances in projects in which Sumit does not have project permissions. For example, if Sumit clicks **Filter** , selects a project on which he does not have any permissions, and clicks **Apply**, the following error is displayed:

User Sumit does not have sufficient privilege to perform this action.

- h. Clicks **Actions ...** to view the instance statistics for an integration instance that is part of the **HCM Project12** project.



- i. Selects the **Edit schedule** option, but receives the following error:

User Sumit does not have sufficient privilege to perform this action.

The restriction occurs because Sumit's monitor permissions do not support this task.

- j. In the navigation pane, clicks the **Instances** tab to view all tracking instances except for instances in projects in which Sumit does not have project permissions.
- k. Performs monitoring tasks successfully:
 - Views the activity stream.

- Aborts and retries instances.
- I. In the navigation pane, clicks the **Errors** tab to view all instances in error except for instances in projects in which Sumit does not have project permissions.
- m. Performs monitoring tasks successfully:
 - Views the activity stream.
 - Aborts and retries instances.

Create or Import a Project

You can create a project in which to design, manage, and monitor integrations. You can also import user-developed, accelerator, or recipe projects into an instance.

- [Create or Import a Project](#)
- [Understand an Imported Oracle Accelerator Project](#)
- [Add a Newer \(Extended\) Version of an Accelerator Project](#)
- [Understand an Imported Recipe Project](#)

See Get Started with Integration Recipes and Accelerators in *Getting Started with Oracle Integration 3*.

Create or Import a Project

1. In the navigation pane, click **Projects**.
2. Click **Add**.
3. Select an option:
 - Create a new project.
 - Import an existing project.
4. To create a new project, perform the following steps.
 - a. Click **Create**.
 - b. Enter the following information.

Field	Description
Name	Provide a meaningful name so that others can understand the project. You can include English alphabetic characters, numbers, underscores, and dashes in the identifier. Note: You cannot create a project name containing <code>-BA-</code> or <code>_BA_</code> . These conventions are reserved for accelerator projects.
Identifier	Accept the default identifier value. The identifier is the same as the project name you provided, but in upper case.
Keywords	Enter keywords (tags) to identify the project. You can search for projects on the Projects page using keywords. Keywords are useful for filtering and discovery.
Description	Provide a meaningful description so that others can understand the project.

Field	Description
Anyone can edit, view, and monitor	<ul style="list-style-type: none"> Select this option to allow anyone to edit, view, and monitor resources in the project. You can always restrict access at a later time. Don't select this option if you want to restrict access to editing, viewing, and monitoring resources in this project. After project creation, you can select the specific groups and users to which to provide access. See Control Who Can Edit, View, and Monitor in a Project.

c. Click **Create**.

Your new project page is displayed. You can now add integrations, connections, lookups, and JavaScript libraries to the project.

5. To import an existing project (.CAR file), perform the following steps. This enables you to import a project with all assets and metadata from one system to another. You can import user-developed, accelerator, or recipe projects.

Ensure that you understand connection behavior when importing a project. See [Export a Project](#).

a. Click **Import**.

b. Click the field to browse for a project or drag and drop a project into the field.

c. With the **Anyone can edit, view, and monitor** check box, select the level of access you want to provide to other users:

- Select this option to allow anyone to edit, view, and monitor resources in the project. You can always restrict access at a later time.
- Don't select this option if you want to restrict access to editing, viewing, and monitoring resources in this project. After import, you can select the specific groups and users to which to provide access. See [Control Who Can Edit, View, and Monitor in a Project](#).

d. Click **Import**.

The project connections contain information that is specific to the processing environment. Connections in the imported project omit this potentially sensitive data. You must reconfigure connections before they can operate properly.

Understand an Imported Oracle Accelerator Project

If you import an Oracle accelerator project into your project, the project includes the labels **Accelerator** and **Oracle** on the Projects page. The **Type** column also includes the **Accelerator** label.

Name	Type	Last updated	Status
Test Primavera OIC Project BA NA <small>Accelerator ORACLE</small>	Accelerator	18 Jan 2023 06:27:12 PM GMT	Configured

Note the following accelerator project restrictions:

- You cannot edit the details that were specified when the project was created, such as name, project identifier, description, and keyword selections. These details are in read-only mode.

- The **Edit**, **Clone**, and **Delete** tasks are not visible under the **Actions ...** menu on the **Deploy** tab. The only task visible under this menu is **View**.
- You can extend (customize) an integration in the accelerator project by selecting **Actions ...**, then **Extend**. See [Extend an Integration in an Accelerator Project](#).

Note the following restrictions for individual assets in accelerator projects

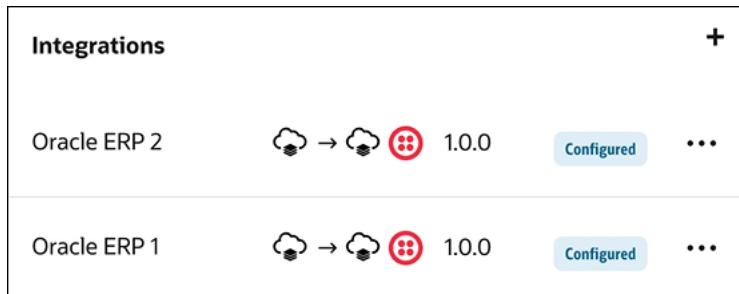
Integrations	Connections	Lookups	JavaScript Libraries	Deployments	Certificates
Allowed:	Allowed:	Allowed:	Allowed:	Allowed:	No restrictions.
<ul style="list-style-type: none"> View Activate Deactivate Override default property values 	<ul style="list-style-type: none"> View Test Refresh metadata Edit and save Save and reactivate 	<ul style="list-style-type: none"> View Edit and save Add or remove rows and update values in rows 	<ul style="list-style-type: none"> View Add libraries Delete libraries 	<ul style="list-style-type: none"> Uninstall View Export ** Activate Deactivate 	
Restricted:	Restricted:	Restricted:		Restricted:	
<ul style="list-style-type: none"> Add Delete Clone Create new version 	<ul style="list-style-type: none"> Add connection s Delete connection s Update primary information 	<ul style="list-style-type: none"> Add lookups Delete lookups Update domain name Add or remove domain Update primary information 		<ul style="list-style-type: none"> Create, update, or delete the deployment 	
Allowed in extended accelerator:					
<ul style="list-style-type: none"> Add a scope or stitch action before an invoke Merge a scope or stitch action from the next lower version of the integration 					
Restricted in extended accelerator:					
<ul style="list-style-type: none"> Update primary information 					

** Export as an accelerator or extended accelerator only.

Add a Newer (Extended) Version of an Accelerator Project

Assume a newer version of an accelerator project becomes available for you to import.

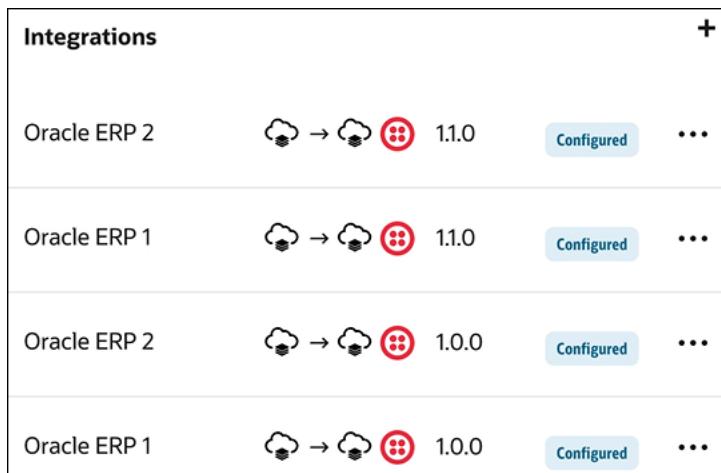
For this example, assume the first version of the accelerator project included integrations Oracle ERP 1 and Oracle ERP 2, each with versions of 1.0.0.



The two integrations and their versions are included in a project deployment named 1.0.0 that you previously imported into your instance.

Name	Identifier	Integrations	Last updated
1.0.0	01.00.0000	2	18 Jan 2023 06:26:27 PM GMT

A newer version of the accelerator project then becomes available. This version now includes two versions of the two integrations: Oracle ERP 1 versions 1.1.0 and 1.0.0 and Oracle ERP 2 versions 1.1.0 and 1.0.0.



All four integrations and versions are included in a project deployment named 1.1.0.

Test Primavera OIC Project BA NA			
			Design Deploy Observe
			Export Activate Deactivate
Name	Identifier	Integrations	Last updated
1.0.0	01.00.0000	2	18 Jan 2023 06:26:27 PM GMT
1.1.0	01.01.0000	2	18 Jan 2023 06:31:55 PM GMT

1. In the navigation pane, click **Projects**.
2. Click **Add**.
3. Click **Import** and upload the latest accelerator project.
The following message is displayed:

Project has been successfully imported.

Name	Type	Last updated	Status
Test Primavera OIC Project BA NA Accelerator ORACLE	Accelerator	18 Jan 2023 06:27:12 PM GMT	Configured

4. Click the project name or click .
Note that the latest accelerator project includes Oracle ERP 1 versions 1.1.0 and 1.0.0 and Oracle ERP 2 versions 1.1.0 and 1.0.0.

Integrations					
Oracle ERP 2	→		1.1.0	Configured	...
Oracle ERP 1	→		1.1.0	Configured	...
Oracle ERP 2	→		1.0.0	Configured	...
Oracle ERP 1	→		1.0.0	Configured	...

Understand an Imported Recipe Project

When you import a recipe project, it is displayed as **Developed** in the **Type** column on the Projects page.

Name	Type	Last updated	Status
Oracle R Primavera Sync	Developed	18 Jan 2023 06:33:37 PM GMT	Configured

Create, Import, or Copy an Integration into a Project

You can create, import, or copy integrations into a project. The integration to copy can be either in a different project (as long as it is marked as globally available) or outside of any project.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, click **Add** if no integrations currently exist or **+** if integrations already exist.
The Add integration panel opens.
4. Select an option to add an integration.

Option	Description
Create	Select to create one of the following integration patterns: <ul style="list-style-type: none"> • Application • Schedule • Event Follow the steps to create an integration. See Create an Integration .
Import	Select to import an integration (.IAR) file. See Import an Integration . Note the following details: <ul style="list-style-type: none"> • You can import an integration that was not created in a project (for example, it was created as a standalone integration in another instance). • You can import integrations created outside of a project that publish and subscribe to events. Importing a publishing integration into a project also imports the event. The event becomes visible in the Events section of the project details page.

Option	Description
Copy	<p>You can copy an integration into a project. The integration to copy can be available in either of the following locations:</p> <p>In another project.</p> <p>a. Ensure that the Search in projects check box is selected. This is the default selection.</p> <p>b. Search for or select the project that includes the integration to copy. Besides searching the listed projects, all available projects in Oracle Integration are searched on the basis of full name, partial name, keywords, and description to fetch the data from the server side.</p> <p>c. Search for or select the integration to copy in that project. Only the listed integrations are searched. That is, the integrations exposed outside of the selected project are listed upon selection of the project. (Data is only fetched from the client side.)</p> <p>Even though the integration to copy is in a project, it must be publicly available. That is, you selected the Available to other projects check box during integration creation or editing under Primary Info  in the integration canvas. Otherwise, you receive the following error:</p> <p style="text-align: center;">Either there is no integration available or visible outside the project.</p> <p>Integrations in projects in which the Available to other projects check box is <i>not</i> selected are not available for selection.</p> <p>d. Click Copy.</p> <p>Globally available (that is, outside any projects). Besides searching for listed integrations, all globally available integrations in Oracle Integration are searched on the basis of full name, partial name, keywords, and description.</p> <p>a. Deselect the Search in projects check box.</p> <p>b. Search for or select the integration to copy.</p> <p>c. Click Copy.</p>

5. Follow the instructions to design your integration type:

- See [Create Application Integrations](#).
- See [Create Schedule Integrations](#).
- See [Create Event Integrations](#).

Unlock an Integration in a Project

You can unlock and recover unsaved changes in an integration in a project. An integration becomes locked if your browser fails during design-time.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, hover over the specific integration to unlock.
4. Click **Actions ...** and select **Unlock**.
5. Follow the steps to unlock and recover unsaved changes in an integration. See [Recover Unsaved Integration Changes](#).

Refresh Endpoints of an Integration in a Project

You can refresh the endpoints of an integration in a project that includes endpoints that support refreshing artifacts.

1. In the navigation pane, click **Projects**.
 2. Click the project name or click .
 3. In the **Integrations** section, hover over the specific integration to refresh.
 4. Click **Actions ...** and select **Refresh Endpoints**.
 5. Select the connections from the list for which to refresh artifacts. Refreshing endpoints may impact the business identifiers and maps.
 6. Click **Refresh**.
- Or:
1. In the **Integrations** section, click a specific integration name or click **Edit** .
 2. In the integration canvas, click **Refresh endpoints** .
 3. Select the connections from the list for which to refresh artifacts. Refreshing endpoints may impact the business identifiers and maps.
 4. Click **Refresh**.

Update Property Values of an Integration in a Project

You can update property values of an integration in a project. At runtime, these integration properties take effect and override the values you configured at design time.

Note the following guidelines for viewing and editing property values in integrations.

Integration Pattern	When Activated	When Deactivated
Application integration	Can view, but not edit.	Can edit.
Schedule integration	Can edit.	Can view, but not edit.

1. In the navigation pane, click **Projects**.

2. Click the project name or click .
3. In the **Integrations** section, hover over the specific integration to update.
4. Click **Actions ...** and select **Update property values**.

See [Override Design-Time Properties in an Integration](#).

Create a Version of an Integration in a Project

You can create a new version of an integration in a project.

Note:

Integration versions follow a formatting convention of `xx.yy.zzzz`, where `xx` is the major version and `yy.zzzz` is the minor version. If you clone an integration (for example, version `1.00.0000`) and change the minor version of the cloned integration to `1.10.0000`, version `1.00.0000` is deactivated when you activate version `1.10.0000`. To keep both integrations active, change the major version of the cloned integration to `2.00.0000`. This enables integration versions `1.00.0000` and `2.00.0000` to be active at the same time.

1. In the navigation pane, click **Projects**.
 2. Click the project name or click .
 3. In the **Integrations** section, find the integration to version.
 4. Click **Actions ...**, then select **Create a new version**.
- The Create new version panel opens.
5. Enter a name, unique identifier, version number, keywords, 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 **Create version**.

Add and Share a Connection Across a Project

You can add a connection to use in an integration in a project. During creation, you can also select to make the connection available across projects. This selection allows you to share the connection resources from one project with other projects. This feature also eliminates the need to create separate connections in each project.

- [Create a Connection](#)
- [Share Connection Resources Across Projects](#)
- [Refresh Endpoints Using Connection Sharing](#)
- [Share Connection Resources in Accelerator Projects](#)

Create a Connection

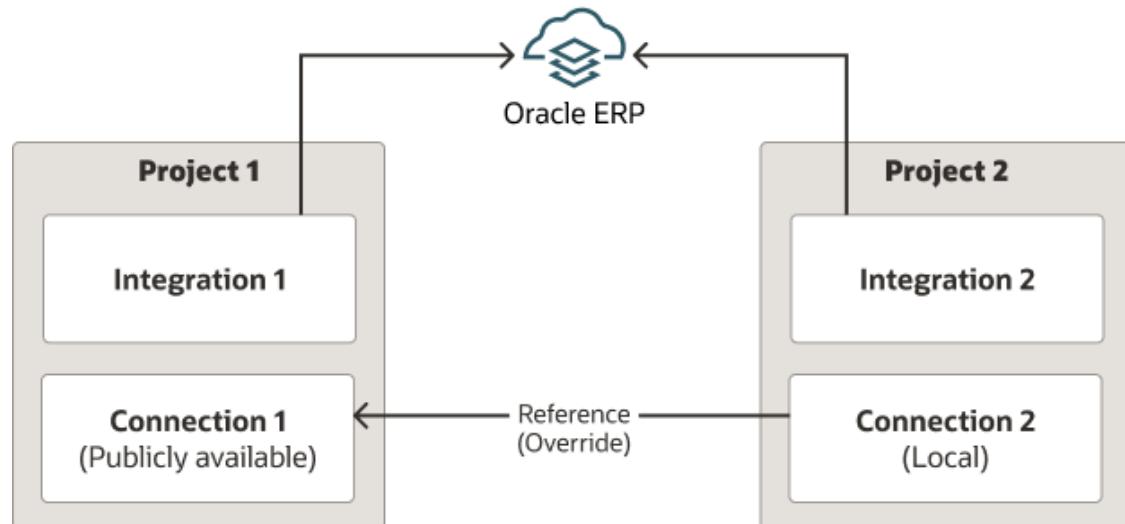
1. In the navigation pane, click **Projects**.

2. Click the project name or click .
3. In the **Connections** section, click **Add** if no connections currently exist or **+** if connections already exist.
The Create connection panel opens.
4. Select the adapter to use in the connection.
5. Follow the steps to create a connection. See [Create a Connection](#).
The connection property and connection security values are specific to each adapter. Your connection may also require configuration with an access type such as a private endpoint or an agent group. See [Configure Connection Properties and Security Properties](#) and [Create a Connection with an Agent Group](#).
6. Test the connection. See [Test the Connection](#).

Share Connection Resources Across Projects

You can share adapter connection resources across projects. For example, you may have two projects that need to integrate with a common system, such as Oracle ERP Cloud. You create and configure the following connection types:

- Publicly available connection: You create, configure, and maintain a single Oracle ERP Cloud Adapter connection that you select to share with other projects. This selection makes the connection publicly available to adapters of the same adapter type and role in other projects.
- Local connections: You create additional connections in other projects that reference the publicly available connection. Referencing the connections means that you don't explicitly configure connection properties, security properties, or access types for these connections. You override the local connection settings by referencing the settings of the publicly available connection. This reference enables you to share resources across each project.



 **Note:**

Note the following connection sharing restrictions:

- If you update the connection configuration settings for a publicly available connection, the activated integrations for any local connections that are referencing the publicly available connection must be reactivated to capture those updates.
- Lookup and JavaScript library resources cannot be shared across projects.
- You cannot track failed integration instances by the publicly-available connection.

The following example describes how to configure connection sharing:

- You create and configure a publicly available connection.
 - You create a local connection that references the configuration settings of the publicly available connection.
1. Open a project.
 2. In the **Connections** section, click **Add** if no connections currently exist or **+** if connections already exist.
 3. Create an adapter connection. For this example, a REST Adapter is selected. See [Create a Connection](#).
 4. Select a role from the **Role** list (for this example, **Trigger** is selected) and click **Share with other projects** to make this connection publicly available for use across projects.

Create connection

REST adapter

Name: testConn

Identifier: TESTCONN

Role: Trigger

Keywords:

Description:

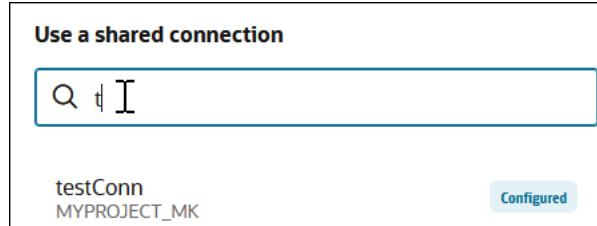
Share with other projects

< Create

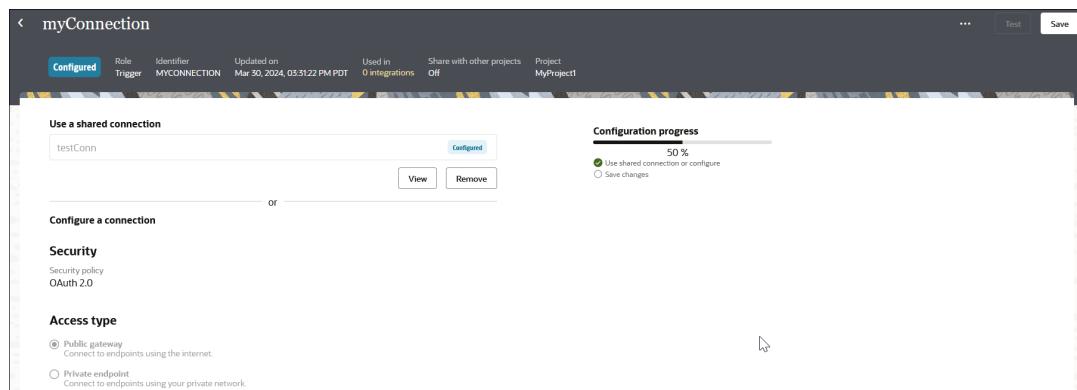
5. Configure the connection properties, connection security, and access control for the publicly available adapter.
6. Test and save the connection, then exit the Connections page.
The connection is displayed as **Configured** in the project.

Connections		+
rest testConn	Trigger	Configured ...

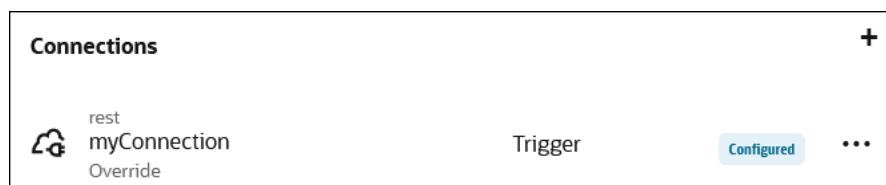
7. Open a different project to create the local connection.
8. Create a connection of the same adapter type (REST Adapter) and same role (**Trigger**) to match the publicly available adapter.
The Connections page is displayed. Note the **Use a shared connection** field at the top.
9. Enter a letter that is part of the name of the publicly available connection (for this example, t is entered).
A filtered list of connections is returned that shows only publicly available connections of the same adapter type (REST Adapter) and same role (**Trigger**) that include that letter.



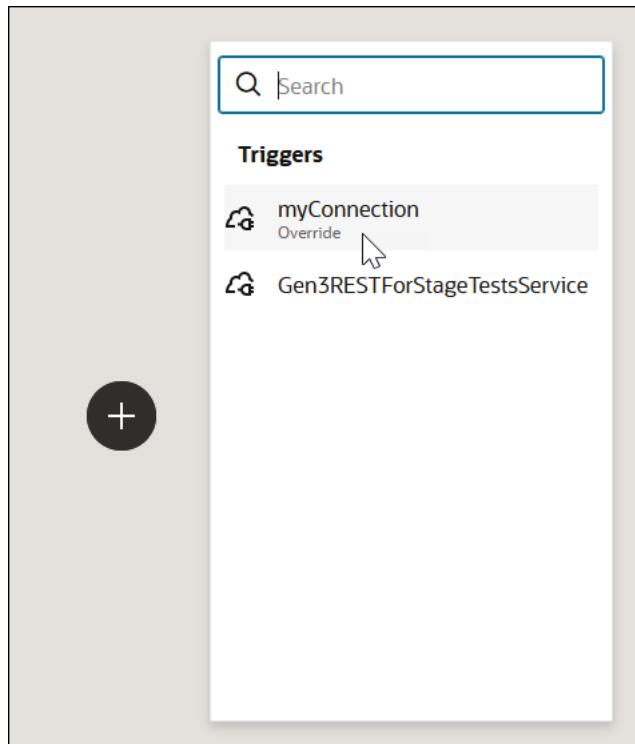
10. Select the publicly available connection (for this example, **testConn**). If the project of the publicly available connection is configured with role-based access control (RBAC) settings, you must have at least view permissions in that project.
11. Click **Save**.
The local connection automatically references the connection configuration settings of the publicly available connection.



12. Save the connection, then exit the Connections page.
The connection is displayed as **Configured** in the project. The **Override** label is displayed, meaning this connection references the resources of the publicly available connection.



13. Create an integration.
The local connection is available for selection in the integration canvas. The **Override** label indicates that this connection is referencing the resources of the publicly available connection in the other project.

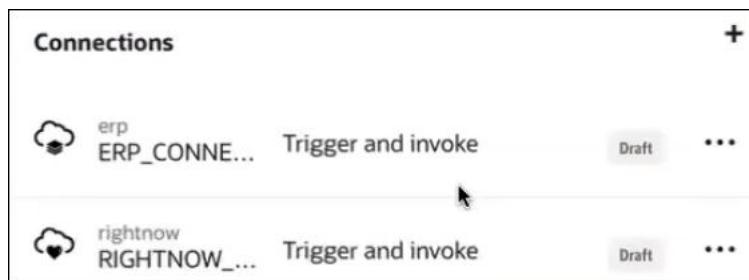


14. Design the remaining parts of your integration.
15. Activate and run the integration.
16. View the activity stream and note that the integration instance was run by referencing the resources of the publicly available connection.

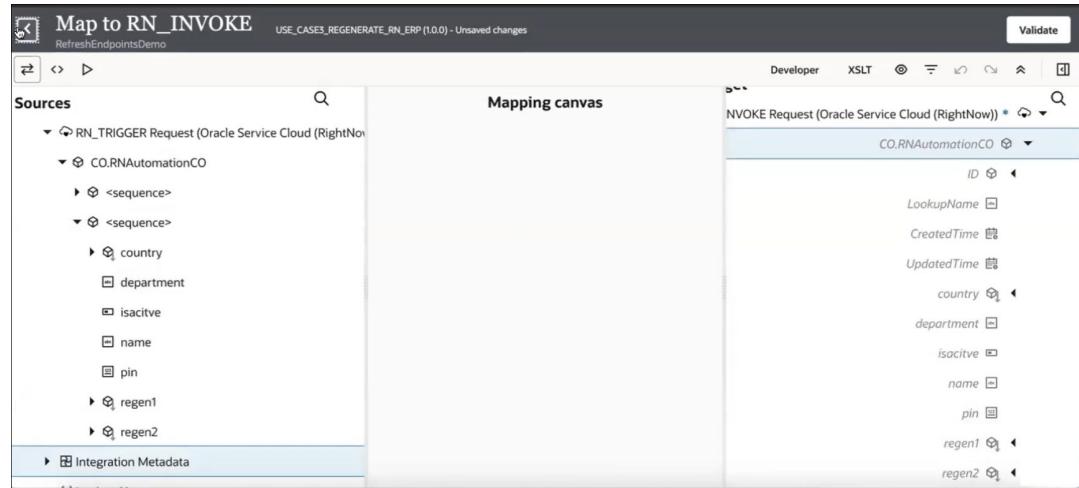
Refresh Endpoints Using Connection Sharing

You can override a local connection in a draft state with a publicly-available connection in a configured state, and then regenerate the endpoints for the local connection. This section describes this capability with a brief example.

1. Create a local connection and leave it in a draft state (for this example, the connection is named **RIGHTNOW_CONN**).



2. Go to the integration canvas and view the map action for the connection. This shows the contents of the mapper prior to endpoint refreshing. Note that there are source and target elements named **regen 1** and **regen 2**.



3. Return to the **Connections** section and open the connection (**RIGHTNOW_CONN**) for editing.
4. In the **Use a shared connection** field, enter the beginning letter to filter the list, and select the shared connection to use (for this example, named **right_NOW**).

Identifier	Role	Updated on
RIGHTNOW_CONN	Trigger and invoke	May 29, 2024, 05:02:02 PM

Use a shared connection

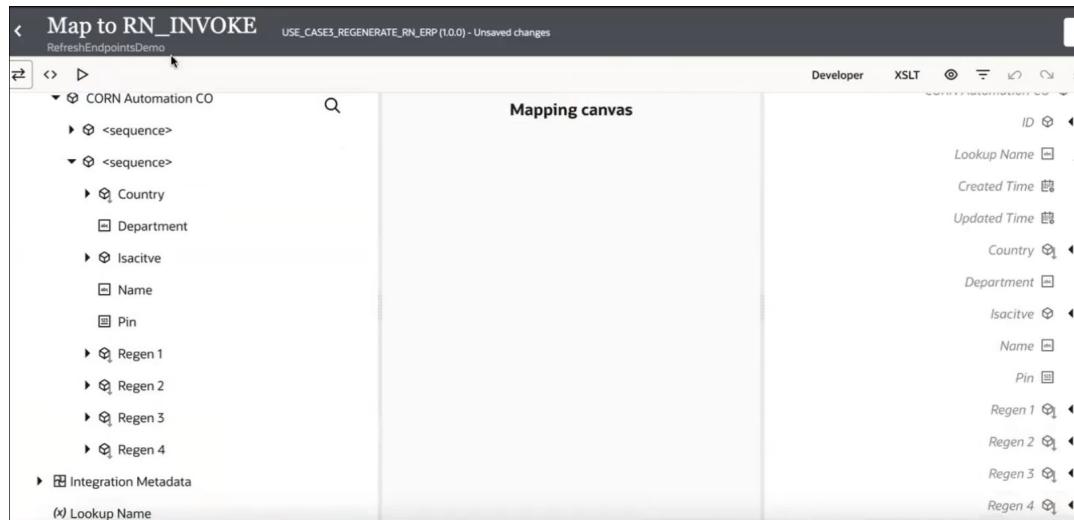
Identifier	Role	Status
REGENERA_PROJECT_3_APR	Trigger and invoke	Draft
RightNOW_CONN3 REGENERATI_PROJECT_3APR	Trigger and invoke	Draft
RightNOW_CONN2 REGENERATI_PROJECT_3APR	Trigger and invoke	Draft
right_NOW MANUAL_TEST	Trigger and invoke	Configured

Connection identifier: RIGHT_NOW

5. Save the connection.
The connection state is changed to configured.

The screenshot shows the configuration page for a connection named "RIGHTNOW_CONN". The top bar indicates the connection is "Configured". Key details shown are Role: "Trigger and invoke", Identifier: "RIGHTNOW_CONN", and Updated on: "May 29, 2024, 05:02:02 PM". Below this, there are two main sections: "Use a shared connection" (with a dropdown menu showing "right_NOW" and a "View" button) and "Configure a connection" (with a "Properties" section containing a "WSDL URL" field). Further down is a "Security" section with fields for "Security policy", "Username", and "Password Token".

6. Go to the **Integrations** section of the project.
7. Hover over the row of the integration, and select **Actions ...**, and then **Refresh endpoints**.
The Refresh endpoints panel opens.
8. Select the connection to refresh (**RIGHTNOW_CONN**), then click **Refresh**.
9. Wait for the refresh to complete.
10. Return to the mapper for the connection. Note that there are now source and target elements named **Regen 1**, **Regen 2**, **Regen 3**, and **Regen 4**.



- 11.** Validate the mappings.
- 12.** Save and exit the integration.

Share Connection Resources in Accelerator Projects

When you install an accelerator project, all connections are initially in a draft state. You can use connection sharing for these connections instead of configuring each connection individually.

For example, if the accelerator project is using Oracle ERP Cloud Adapter and Twilio Adapter connections, you can use connection sharing if the following conditions are met:

- There are publicly available connections for Oracle ERP Cloud Adapter and Twilio Adapter.
- The roles are the same.
- You have view permissions at a minimum on the project that includes the publicly available connections.

After you configure connection sharing for these adapters, you can activate the integration in the accelerator project.

Edit a Connection in a Project

You can edit an existing connection in an integration in a project.

- 1.** In the navigation pane, click **Projects**.
- 2.** Click the project name or click .
- 3.** In the **Connections** section, find the adapter connection to edit.
- 4.** Click , then click **Edit**.

The page for editing the specific adapter's connection properties, security policy, and (if required) agent group is displayed.

- 5.** Follow the steps to edit the connection. See [Configure Connection Properties and Security Properties](#) and [Create a Connection with an Agent Group](#). The connection property and connection security values are specific to each adapter.
- 6.** Test the connection. See [Test the Connection](#).

Refresh Connection Metadata in a Project

You can manually refresh the currently-cached metadata available to adapter connections in a project that have implemented metadata caching.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Connections** section, find the specific adapter connection to refresh.
4. Click **Actions ...** and select **Refresh metadata**.

If successful, the following message is displayed.

Metadata refresh for connection *connection_name* has been initiated successfully.

See [Refresh Integration Metadata](#).

Delete a Connection in a Project

You can delete an adapter connection in a project. A connection currently used in an active integration cannot be deleted. Deleting the connection would invalidate the integration. Active integrations would stop running. Remove the connection from the integration and then delete the connection.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Connections** section, find the specific adapter connection to delete.
4. Click **Actions ...** and select **Delete**.
5. Confirm your selection when prompted.

Unlock a Connection in a Project

You can unlock a connection in a project. A connection becomes locked when the browser fails in edit mode, which prevents it from being edited.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Connections** section, find the specific adapter connection to unlock.
4. Click **Actions ...** and select **Unlock**.

See [Unlock a Connection](#).

Add a Lookup to a Project

You can create new or import existing lookups in a project. Only the lookups within the project can be used by expressions and in the mapper.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Lookups** section, click **Add** if no lookups currently exist or  if lookups already exist. The Add lookup panel opens.
4. Select an option:
 - Create a new lookup
 - Import a lookup
5. To create a new lookup, perform the following steps.
 - a. Click **Create**.
 - b. Enter the following information.

Field	Description
Name	Provide a meaningful name so that others can understand the lookup. You can include English alphabetic characters, numbers, underscores, and dashes in the identifier.
Keywords	Enter keywords (tags) to identify the lookup. You can search for lookups using keywords.
Description	Provide a meaningful description so that others can understand the lookup.

- c. Follow the steps to add adapters or domain names to a lookup. See [Create a Lookup](#).
- d. Click **Create**.
6. To import an existing lookup, perform the following steps.
 - a. Click **Import**.
 - b. Click the field to browse for a lookup (.csv file) or drag and drop a lookup into the field.
 - c. Click **Import**.

See [Import a Lookup](#).

Edit a Lookup in a Project

You can edit a lookup in a project.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Lookups** section, find the lookup to edit.
4. Click the lookup name or click **Actions ...** and select **Edit**.

See [Manage Lookups](#).

Clone a Lookup in a Project

You can clone a lookup in a project and customize its content to use in other integrations. Cloning provides a quick way to create a new lookup with similar information.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Lookups** section, find the lookup to clone.
4. Click **Actions ...** and select **Clone**.
5. Enter a name, optional keyword, and optional description.

Delete a Lookup in a Project

You can delete a lookup in a project. A lookup cannot be deleted if in use by an integration. Deleting the lookup would invalidate the integration and cause an active integration to stop running. You must first remove the lookup from the integration and then delete the lookup.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Lookups** section, find the lookup to delete.
4. Click **Actions ...** and select **Delete**.

See [Delete a Lookup](#).

Import a JavaScript Library in a Project

You can import a JavaScript library file in **.ZIP**, **.JAR**, or **.JS** format into a project. Only those libraries within the project can be used by JavaScript callouts.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Libraries** section, click **Add** if no libraries currently exist or **+** if libraries already exist.
The Import panel opens.
4. Click the field to browse for a library file or drag and drop a library file into the field.

See [Import a Library File](#).

Delete a JavaScript Library in a Project

You can delete a JavaScript library in a project. If any of the library's functions are used by an integration, you are prevented from deleting the library. You must remove the functions from the integration before you can delete the library.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Libraries** section, find the library to delete.
4. Click **Actions ...** and select **Delete**.

Update a JavaScript Library in a Project

You can update a JavaScript library in a project.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Libraries** section, find the library to update.
4. Click **Actions ...** and select **Update**.

See [Update JavaScript Files](#).

Create, Publish, and Subscribe to Events in a Project

You can create events that you publish and subscribe to in integrations in a project.

The process you follow to create, publish, and subscribe to an event in a project is the same as outside of a project. The only difference is that the events and integrations are only available within that specific project. The following steps provide an overview of the process and cross-reference other sections of this guide for more specific procedures.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. Scroll down to the **Events** section at the bottom.
4. In the **Events** section, click **+** to create an event.
The Define new event panel opens.
5. Create the event payload structure. See Step 3 in [Create Events to Publish and Subscribe to in Integrations](#).
6. After you create events, you can view and perform additional actions in the **Events** section of the projects.

The screenshot shows a list of events in a project. The events listed are:

- test_im
- verify_proj_evt1
- VerificationEvent
- test_gimishra

Each event entry has a three-dot menu icon to its right. At the top right of the list area is a large plus sign icon. At the bottom left of the list area is a link labeled "View all event(s)" with a small arrow icon.

- Click **Actions ...** to access a menu with the following options:

Option	Description
Clone	Enables you to clone the event.
Edit	Enables you to edit the event payload structure.
Delete	Enables you to delete an event. You cannot delete an active event.
View	Enables you to view the event payload structure, the integrations publishing the event, and the integrations subscribing to the event.

- Go to the **Integrations** section of the project.
 - Create an application or schedule integration to publish the event. During integration design, you drag a **Publish event** action into the integration canvas and select the event to publish. See [Publish Events in an Integration](#).
 - Return to the **Integrations** section of the project.
 - Create an event integration to subscribe to the event. You must select an **Event** integration pattern. During integration design, you select the event to which to subscribe. Only the events available within that project are available for selection. See [Subscribe to Events in an Integration](#).
- The subscribing integration event is triggered when the integration publishing the event is run. Multiple integrations can subscribe to the same published event.

Create Robots in a Project

The only place to build robots and their related components is in a project.

All tasks related to robots and robotic process automation are documented in a separate guide. The following table provides links to this guide.

Goal	More information
Enable robots in your instance	Start Building Robots Today! in <i>Using Robots in Oracle Integration 3</i>
Learn about robots	Learn About Robots in <i>Using Robots in Oracle Integration 3</i>
Build a robot	Quick Start for Building Robots in <i>Using Robots in Oracle Integration 3</i>

Create Healthcare Integrations in a Project

You can create HL7 messages, create schemas, import FHIR profile packages, and create integrations using the healthcare action in a project if your Oracle Integration instance includes the Healthcare edition.

This table provides an overview of healthcare tasks in a project and provides references to relevant documentation.

Functionality	See
Install or upgrade to the Healthcare edition. To create healthcare integrations in a project, your Oracle Integration instance must include the Healthcare edition.	<ul style="list-style-type: none"> Create an Oracle Integration Instance in <i>Provisioning and Administering Oracle Integration 3</i> Edit the Edition, License Type, Message Packs, and Custom Endpoint of an Instance in <i>Provisioning and Administering Oracle Integration 3</i>
You can perform the following healthcare-related tasks in projects: <ul style="list-style-type: none"> Create, clone, edit, view, and delete HL7 messages and schemas. Select the HL7 message when configuring the healthcare action in an integration in the Integrations section of the project. Import FHIR profile packages. Export a project that includes healthcare resources and import that project into another instance. Clone a project that includes healthcare resources. Control the users and groups that edit, view, and monitor a project that includes healthcare resources with role-based access control (RBAC). View HL7 messages and healthcare schemas in the dependency diagram. 	<ul style="list-style-type: none"> Tour of the Projects Workspace Create Oracle Integration for Healthcare Schemas and Documents in <i>Using Oracle Integration for Healthcare in Oracle Integration 3</i> Import FHIR Profile Packages in <i>Using Oracle Integration for Healthcare in Oracle Integration 3</i> Convert HL7 Messages with a Healthcare Action in <i>Using Oracle Integration for Healthcare in Oracle Integration 3</i>

Create B2B Integrations in a Project

You can create B2B messages, schemas, and integrations using the B2B action in a project.

This table provides an overview of B2B tasks in a project and provides references to relevant documentation.

Functionality	See
<p>You can perform the following B2B-related tasks in projects:</p> <ul style="list-style-type: none">• Create, clone, edit, view, and delete B2B documents and schemas.• Select the B2B document when configuring the B2B action in an integration in the Integrations section of the project.• Create trading partners, define transports, create agreements, and export and import a trading partner.• Export a project that includes B2B resources and import that project into another instance.• Clone a project that includes B2B resources. <p>Not all B2B features are currently supported in projects. See Design B2B Integrations in Projects or in Standalone Environments.</p>	<ul style="list-style-type: none">• Tour of the Projects Workspace• B2B Schemas and B2B Documents in <i>Using B2B for Oracle Integration 3</i>• Manage B2B Trading Partners in <i>Using B2B for Oracle Integration 3</i>• Use the B2B Action in Standalone Mode in <i>Using B2B for Oracle Integration 3</i>

Manage a Project

You manage a project from a single workspace, including creating project deployments, activating and deactivating project deployments or individual integrations, exporting projects, extending integrations in an accelerator project, upgrading accelerator projects, setting tracing levels on integrations, testing integrations, deleting projects, exporting projects, and more.

Topics:

- [Create and Manage a Project Deployment](#)
- [Activate or Deactivate a Project](#)
- [Edit and Replace Dependent Resources in a Project](#)
- [Export a Project](#)
- [Clone a Project](#)
- [Invoke Child Integrations Inside or Outside of Projects](#)
- [Manage Accelerator Projects](#)
- [Activate or Deactivate an Integration in a Project](#)
- [Deploy Integration Endpoints to Oracle Cloud Infrastructure API Gateway](#)
- [Clone an Integration in a Project](#)
- [Update the Tracing Level of Integrations in a Project](#)
- [Test REST Adapter Trigger Connection-Based Integrations in Projects](#)
- [View the Dependent Relationships Between Project Resources](#)
- [View Run Details About Integrations in Projects](#)
- [View Project Status](#)
- [Edit Project Description Details](#)
- [Delete a Project](#)

Create and Manage a Project Deployment

You can create and activate a project deployment consisting of integrations and robots. When you create the project deployment, you select the versions of the integrations and robots to include. You can also perform project deployment management tasks on user-developed deployments, such as editing, cloning, and deleting the deployment.

Understand the Integration Versioning Life Cycle in a Project Deployment

When you create a project deployment, you must select the integrations and their versions to include. For example, you can select integration A/version 1 and integration A/version 2. You cannot select two or more minor versions of the same major version. By default, the latest version of the integration is selected.

The following example describes the life cycle of a project deployment. The following project initially includes two integrations with the following versions prior to creation of the project deployment:

- Project
 - Integration_A
 - * Version 01.00
 - * Version 01.01
 - * Version 02.00
 - Integration_B
 - * Version 01.00
 - * Version 02.00

Because the project is all-inclusive and integrations are versioned, this project can be deployed in a number of different configurations.

The initial project deployment delivery may include the following integrations and versions:

- Project
 - Integration_A
 - * Version 01.00
 - Integration_B
 - * Version 01.00

After updates are made to Integration_A, the project deployment is subsequently delivered as follows:

- Project
 - Integration_A
 - * Version 01.01
 - Integration_B
 - * Version 01.00

At a later time when Integration_A/Version 02.00 is complete, but Integration_B/Version 02.00 requires further development, the project is delivered as follows:

- Project
 - Integration_A
 - * Version 02.00
 - Integration_B
 - * Version 01.00

In this scenario, development of Integration_B/Version 02.00 can continue without disruption, as can delivery of the project (with Integration_B/Version 01.00).

Create a Deployment

1. In the navigation pane, click **Projects**.

2. Click the project name or click .

3. Click the **Deploy** tab.

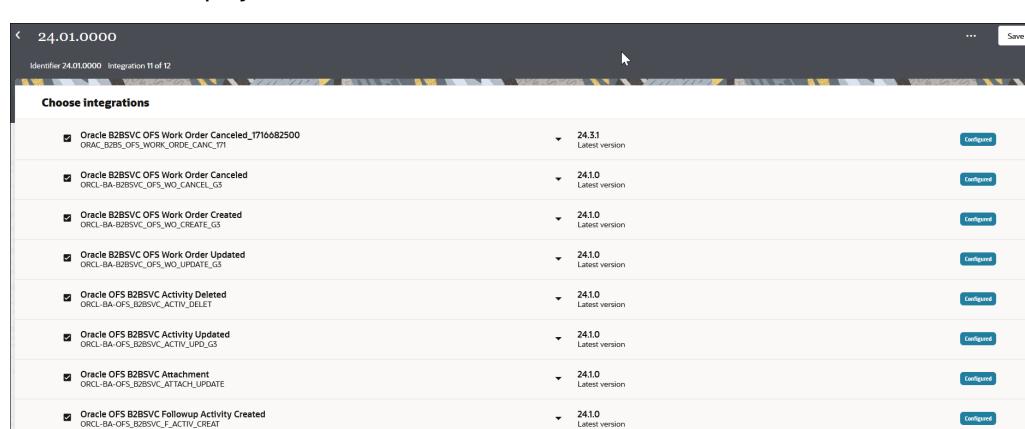
4. Click **Create**.

The Create deployment panel appears.

5. Enter a name, identifier, and optional description. The name is required for activating the project deployment.

6. Select the integrations to include in the deployment.

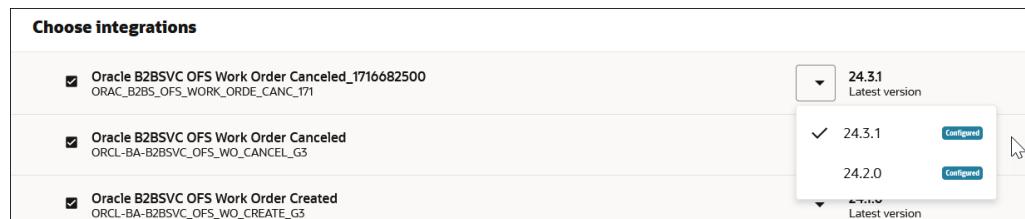
- a. Scroll through the list of integrations included in the project and select the ones to include in the deployment.



Integration	Version	Status
Oracle B2BSVC OFS Work Order Canceled_1716682500	24.3.1	Configured
Oracle B2BSVC OFS Work Order Canceled	24.1.0	Configured
Oracle B2BSVC OFS Work Order Created	24.1.0	Configured
Oracle B2BSVC OFS Work Order Updated	24.1.0	Configured
Oracle OFS B2BSVC Activity Deleted	24.1.0	Configured
Oracle OFS B2BSVC Activity Updated	24.1.0	Configured
Oracle OFS B2BSVC Attachment	24.1.0	Configured
Oracle OFS B2BSVC Followup Activity Created	24.1.0	Configured

The available versions of each integration are displayed in the second column. By default, the most recently-created versions are selected when you create a deployment.

- b. Select the version to include for each integration in the deployment. For this example, select one of the two available versions.



Integration	Version	Status
Oracle B2BSVC OFS Work Order Canceled_1716682500	24.3.1	Configured
Oracle B2BSVC OFS Work Order Canceled	24.3.1	Configured
Oracle B2BSVC OFS Work Order Created	24.2.0	Configured

7. Select the robots to include in the deployment.

Even if you selected an integration that calls a robot, the robot is not automatically included in the deployment. You must manually select all the robots to include.

- a. In the left toolbar, select **Robot** .

- b. Scroll through the list of robots included in the project and select the ones to include in the deployment.

The available versions of each robot are displayed in the second column. By default, the most recently-created versions are selected when you create a deployment.

- c. Select the version to include for each robot in the deployment.

8. Click **Save**, then exit the Choose integrations page.

9. Click **Activate** .

The Activate project panel is displayed. The name of the deployment is displayed automatically and cannot be changed.

10. Select the tracing level:

- **Production**: All activities outside loops and invoke/logger activities inside loops (up to a 1000 iterations) are shown in the activity stream and the data is retained for 32 days.
- **Audit**: In addition to the production settings, wire payloads of trigger/invoke(s) are also shown in the activity stream and the data is retained for eight days.

11. Click **Activate**.

Manage Project Deployments

The **Deploy** tab provides additional management tasks.

1. In the navigation pane, click **Projects**.

2. Click the project name or click .

3. Click the **Deploy** tab.

4. Select **Actions ...** for a user-developed project deployment, and note the following options. For accelerator projects imported into your instance, only the **View** option is visible. See [Understand an Imported Oracle Accelerator Project](#).

Action Option	Description
Edit	Select to edit a project deployment.
View	Select to view a project deployment in read-only mode.
Clone	Select to clone a project deployment.
Export	Select to export a project deployment.
Delete	Select to delete a project deployment. Note: If you delete an individual integration, it is removed from all deployments in which it is included.

Activate or Deactivate a Project

You can activate or deactivate a project. Project activation or deactivation results in bulk processing of the selected integrations in the project.

- [Activate a Project](#)
- [Deactivate a Project](#)

Activate a Project

Before you can activate a project, you must create a project deployment and explicitly select the integrations and their versions to include in the deployment. During activation, each contained asset is visited and activated. It's possible that an asset may already be in the necessary state, in which case no action is taken. Project activation is ideally all or nothing, although those assets activated individually are taken into consideration. Assets can be activated in any order. See [Create and Manage a Project Deployment](#).

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the upper right corner, click **Activate**.
The Activate project panel opens.
4. Select a project deployment name.

Note:

If there are no deployment names available, create a project deployment under the **Deploy** and select the specific integrations and integration versions to include.

5. Select a tracing level:
 - **Production**: All activities outside loops and invoke/logger activities inside loops (up to a 1000 iterations) are shown in the activity stream and the data is retained for 32 days.
 - **Audit**: In addition to the production settings, wire payloads of trigger/invoke(s) are also shown in the activity stream and the data is retained for eight days.
6. Click **Activate**.

Deactivate a Project

A project is deactivated in its entirety. All assets are deactivated regardless of whether an asset was activated within the context of a project or individually. Assets can be deactivated in any order.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the upper right corner, click **Deactivate**.
The Deactivate project panel opens.
4. Click **Deactivate**.

Edit and Replace Dependent Resources in a Project

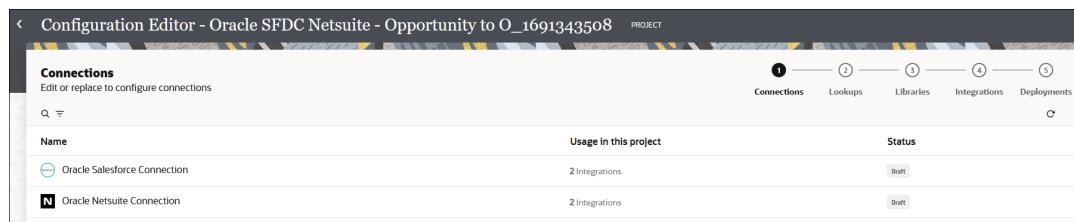
You can edit dependent resources (connections, libraries, lookups, and JavaScript libraries) in a single integration in a project or all integrations in a project with a wizard that's accessible from the Configuration Editor page.

- [Edit Dependent Resources at the Project Level](#)
- [Edit and Replace Dependent Resources at the Integration Level in a Project](#)

Edit Dependent Resources at the Project Level

1. In the navigation pane, click **Projects**.
2. Click **Actions ...**, and select **Configure**.

The Configuration Editor page opens. A wizard is displayed at the top. This wizard guides you through configuration of all potential resources in a project (regardless of whether resources have currently been defined).



3. If you want to edit the connection properties and security properties, hover over the row and click **Edit** .
 - a. Edit the connection.
 - b. Click **Test**, then **Save**.
The active and inactive integrations using the connection are displayed.
 - c. Click **Save** to save your changes or **Save & Reactivate** if the connection you edited is being used in a currently activated integration.
 - d. Click **Back** .
4. Click the **Lookups** icon. A green icon is displayed for the **Connections** icon.
 - a. If you want to edit a lookup, click **Edit** .
 - b. Edit the lookup.
 - c. Click **Save**.
 - d. Click **Back** .
5. Click the **Libraries** icon. A green icon is displayed for the **Lookups** icon.
 - a. If you want to edit a library, click **Edit** .
 - b. Edit the library.
 - c. Click **Save**.

- d. Click **Back** .
6. Click the **Integrations** icon. A green icon is displayed for the **Libraries** icon.
 - a. Hover over the row of an integration.
 - b. If you want to edit the integration, click **Actions ...**, then **Edit**.
 - c. If you want to activate an integration, click **Activate** . You can also deactivate currently activated integrations.
If all connections are configured, the Activate integration panel is displayed. If you have incomplete connections, you are prompted to first configure the inactive connections and try activation again.
 - d. Select a tracing level, then click **Activate**.
7. Click the **Deployments** icon. A green icon is displayed for the **Integrations** icon.
 - a. Hover over the row of the project.
 - b. If you want to edit the deployment, click **Actions ...**, then **Edit**.
 - c. If you want to activate the project deployment, click **Activate** . If you have not created a deployment, you are prompted to create one.

Edit and Replace Dependent Resources at the Integration Level in a Project

1. In the navigation pane, click **Projects**.
2. Click the project name or click **Edit** .
3. In the **Integrations** section, find the specific integration to configure.
4. Click **Actions ...**, and select **Configure**.
The Configuration Editor page opens. A wizard is displayed at the top. This wizard guides you through configuration of all current resources in an integration. If resources have not been defined, the wizard does not display steps for configuring them.
5. If you want to replace a connection with a different one, click **Replace** .
Note the following connection replacement guidelines:
 - Only connections of the same role type (trigger, invoke, or trigger and invoke) are displayed for selection. You can only replace a connection resource with another resource of the same role.
 - The connection status must be displayed as **Configured**. You cannot replace a connection with a connection whose status is **Draft**. If a compatible connection resource does not exist, a message is displayed.
 - You cannot replace connections used in integrations with the following state:
 - Active
 - Activation in progress
 - Deactivation in progress
 - Locked
 - If you replace a connection, a **Revert** icon is displayed in the row that enables you to change back to the previous connection.
6. If you want to edit the connection properties and security properties, click **Edit** .

- a. Edit the connection.

- b. Click **Test**, then **Save**.

The number of active and inactive integrations using the connections are displayed.

- If there are no active integrations, reactivation of integrations is not available.
- If you made changes, click **Save & Reactivate** to apply the connection changes to active integrations. Schedule integrations with an active schedule are excluded. Any queued/in-progress schedule integration instances are canceled.

- c. Click **Back** .

If your integration includes lookups, the **Lookups** icon appears at the top.

7. Click the **Lookups** icon.

- If the connection selected in the previous step is in a configured state, a green icon is displayed for the **Connections** icon.
- If the connection selected in the previous step is *not* in a configured state, an error icon is displayed for the **Connections** icon. Hover over the icon to show information about the error.

- a. If you want to edit a lookup, click **Edit** .
- b. Edit the lookup.
- c. Click **Save**.

- d. Click **Back** .

If your integration includes libraries, the **Libraries** icon appears at the top.

8. Click the **Libraries** icon. A green icon is displayed for the **Lookups** icon.

- a. If you want to edit a library, click **Edit** .
- b. Edit the library.
- c. Click **Save**.

- d. Click **Back** .

9. Click the **Activation** icon. A green icon is displayed for the **Libraries** icon.

- a. Hover over the row of an integration.
- b. If you want to edit a deactivated integration, click **Actions ...**, then **Edit**.
- c. If you want to activate an integration, click **Activate** . You can also deactivate currently activated integrations.
- d. Select a tracing level, then click **Activate**.

See [Edit and Replace Dependent Resources](#).

Export a Project

You can export a project for import into another instance. This action exports any integration resources (connections, lookups, JavaScript libraries, and events) and healthcare resources (HL7 messages, schemas, and integrations using the healthcare action).

Note the following connection behavior when exporting and importing projects.

- A project export artifact (.CAR file) does not include any remote shared connections.
- A local connection's override configuration or association is not exported.
- An imported project does not replace an already existing resource. Therefore, the override configuration on the target system is not lost.

See [Create or Import a Project](#).

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the upper right corner, click **Export** . The Export project panel opens.
4. Select a project deployment name.

 **Note:**

If there are no deployment names available, create a project deployment under the **Deploy** tab and select the specific integrations and integration versions to include. See [Create and Manage a Project Deployment](#).

5. Click **Export**.

Clone a Project

You can clone a project for consumption by other teams that require access to similar integration resources (connections, lookups, JavaScript libraries, and events) and healthcare resources (HL7 messages, schemas, and integrations using the healthcare action). You can also clone an accelerator project to gain full control over its resources. When you clone an accelerator project, it is converted to a developer project.

Note the following project cloning behavior:

- Integrations and their dependent resources such as lookups, connections, JavaScript libraries, and events are cloned.

- Project deployments are cloned.

- Runtime data visible under the **Observe** tab is not cloned.

- Role-based access control (RBAC) settings are not cloned.

- All cloned connections are in a draft state with no connection property or security information defined.

1. In the navigation pane, click **Projects**.

2. Hover over the row of the project to clone.

3. Click **Actions ...**, then select **Clone**.

The Clone project panel opens.

4. Enter a name, unique identifier, optional keyword, and optional description.

5. Click **Clone**.

A message appears indicating your cloning request was submitted.

6. Click **Refresh** .

The cloned version is displayed on the Projects page.

Invoke Child Integrations Inside or Outside of Projects

You can configure a parent integration to invoke a child integration in the same project, a child integration in a different project, or a child integration that is not part of any project (known as an integration that is globally available).

- [Capabilities](#)
- [Restrictions](#)
- [Set a Project Integration as Public](#)
- [Prerequisites to Dynamically Invoke a Child Integration from a Parent Integration After Import into a Project](#)

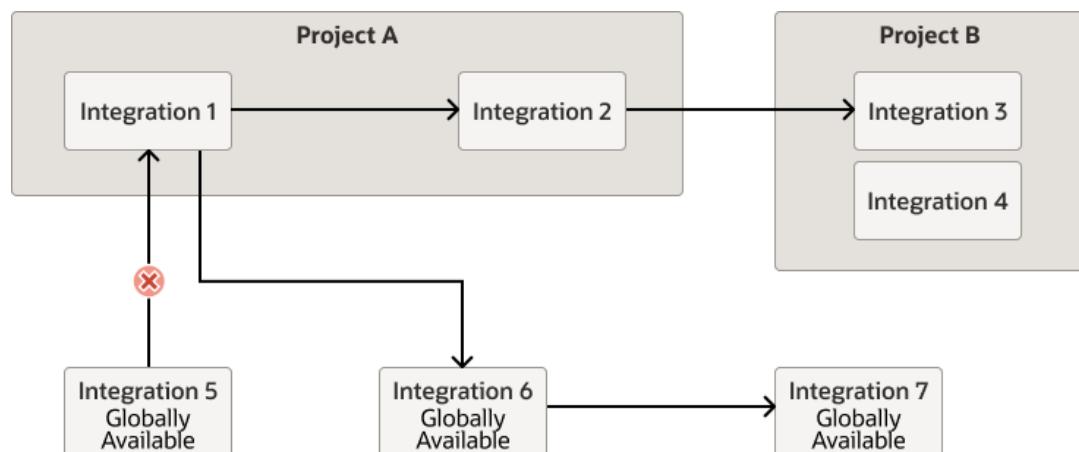
Capabilities

- A parent integration in a project can be configured to invoke a child integration located:
 - In the same project.
 - In a different project, if that integration is set as public. See [Set a Project Integration as Public](#).
 - Globally available (that is, the integration is not included in any project).

During runtime, all integration behavior is consolidated into a single activity stream for viewing, regardless of the location of all the integrations.

As an example, the following parent-to-child invokes are supported:

- Parent Integration 1 in Project A invokes child integration 6 that is globally available (integration is not in any project). Child integration 6 then invokes child integration 7, which is also globally available.
- Parent Integration 2 in Project A invokes child integration 3 in Project B.
- Parent Integration 5, which is globally available, cannot invoke integration 1 in Project A because this type of invocation is not allowed.



- A parent integration in a project can dynamically invoke either of two child integrations at runtime, regardless of location. For example, both child integrations can be in the globally available or one child integration can be globally available and the other child integration can be in another project. See [Dynamically Invoke a Child Integration](#).

Restrictions

- Only REST, SOAP, and schedule integration endpoints are supported.
- Only activated integrations are available when selecting a child integration in the Integration Adapter Wizard. See [Invoke a Child Integration from a Parent Integration](#).
- A parent integration that is globally available (outside a project) *cannot* invoke a child integration inside any project. Parent integrations that are globally available can only invoke child integrations that are also globally available.
- If you select to activate or deactivate a project deployment, all child integrations outside that project (whether in another project or globally available) must already be activated or deactivated, respectively. The project deployment cannot activate or deactivate integrations outside that project. See [Create and Manage a Project Deployment](#).
- When exporting a project deployment, none of the integrations from outside that project are included in the exported file. You *must* separately export those integrations and import them along with the referencing projects into the target instance.
- References to an integration in another project may specify an integration that itself has external references. A chain of references can span multiple projects.

Set a Project Integration as Public

Setting a project integration as public enables that integration to be invoked by a parent integration in a different project. If you don't set a project integration as public, it can only be invoked by a parent integration in the same project.

Set a project integration as public in either of two ways:

Create a new integration:

1. In the **Integrations** section, click **Add** if no integrations currently exist or **+** if integrations already exist.
2. Select the type of integration to create (application or schedule).
3. Select the **Available to other projects** check box.

Update an existing integration:

1. Go to the integration canvas for the integration to make public.
2. Click **Primary Info** .
3. Select the **Available to other projects** check box.

When selected, the **Available to other projects** check box enables an integration in a different project to appear for selection in the Integration Adapter Wizard that is invoked when you drag the **Integration** action into the integration canvas. See [Invoke a Child Integration from a Parent Integration](#).

Prerequisites to Dynamically Invoke a Child Integration from a Parent Integration After Import into a Project

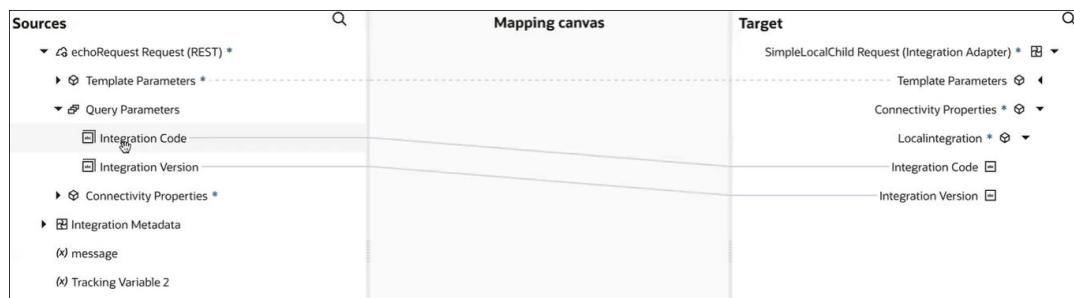
To dynamically invoke a child integration from a parent integration in a project, you need the project ID, integration code, and integration version. This differs from globally-available integrations (integrations not included in any project), where you can dynamically invoke a child integration from a parent integration with only the integration code and integration version.

If you import a globally-available parent integration and its child integrations into a project and then attempt to dynamically call the second child integration from the parent integration, the child integration is not called from the project. Instead, the globally-available child integration is called. This occurs because only the integration code and integration version are available to configure in the mapper, and not the project code.

You must create and map a project code query parameter in the mapper for dynamic invocation of the second integration in a project to succeed.

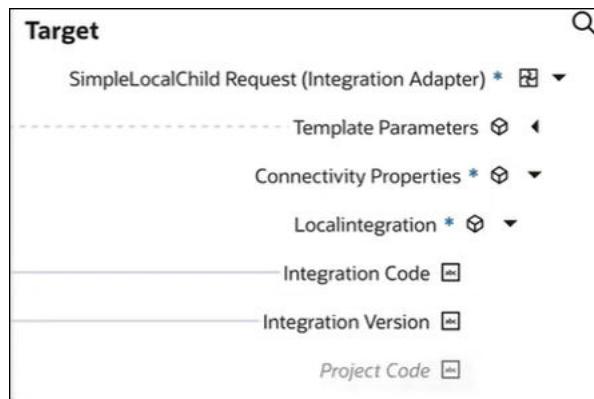
Assume you perform the following steps:

1. Configure a parent integration to invoke two child integrations. All three integrations are globally available, meaning they are not part of a specific project.
2. Export the three integrations and import them into a project.
3. Edit the mapper of the imported parent integration.
Note that there is no value for the target **Project Code** node under **Localintegration** to configure.



Solution: Perform the following steps:

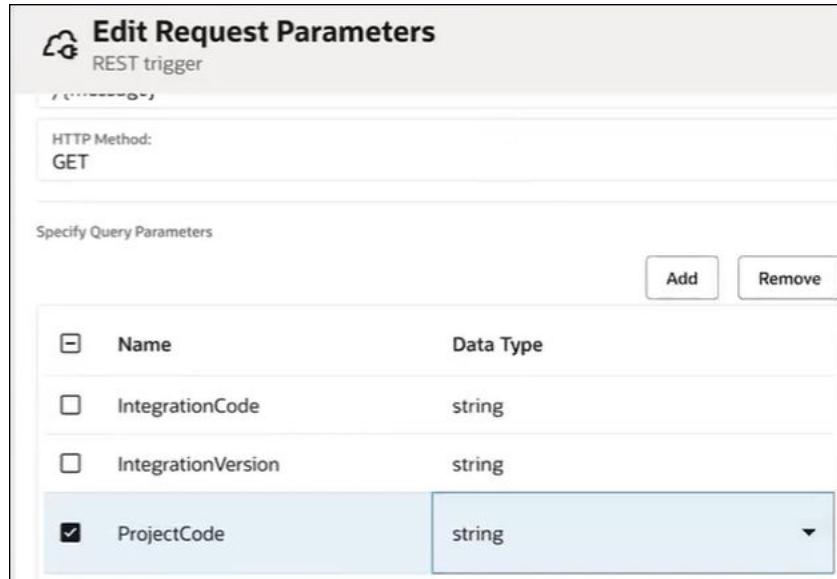
1. In the integration canvas, select to edit the integration action.
2. Click **Continue** on the pages of the Integration Adapter Wizard without making changes, then click **Done** on the final page. This action creates the project code entry in the mapper.
3. Open the mapper and note that the target **Project Code** node for the project under **Localintegration** is now visible.



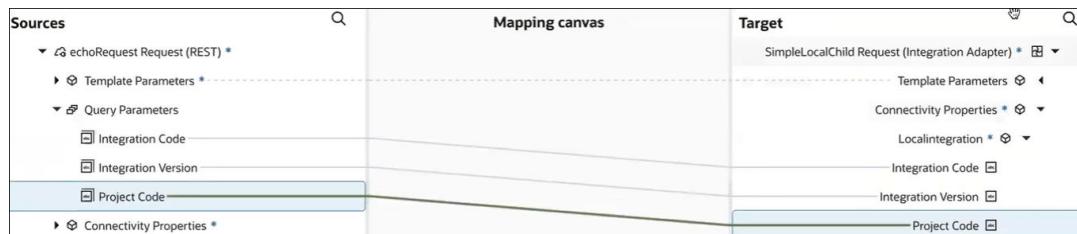
You must now create a target query parameter and map it to **Project Code**.

4. Exit the mapper.

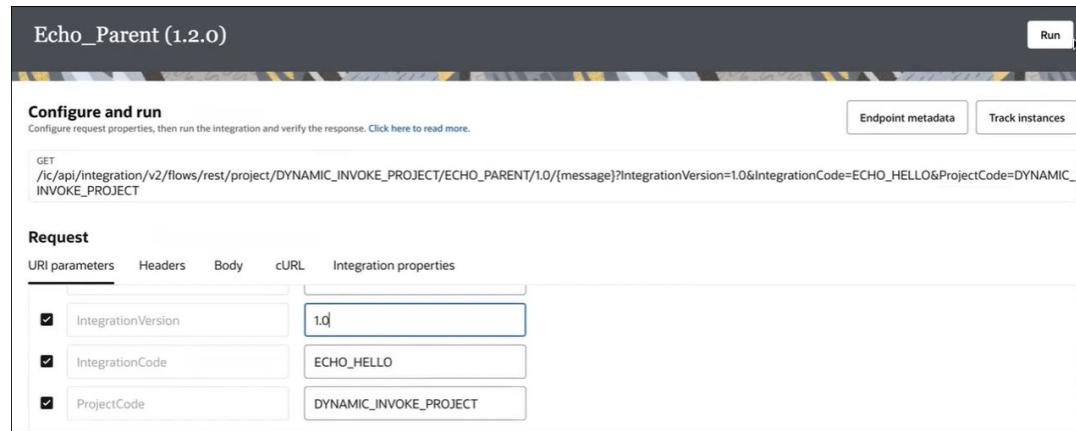
5. Open the REST Adapter trigger connection for editing in the Adapter Endpoint Configuration Wizard.
6. On the Request Parameters page, create a new query parameter. For this example, ProjectCode is created.



7. Click through the remaining pages of the wizard, then click **Done**.
8. Return to the mapper.
The new source query parameter is now visible.
9. Drag the source **Project Code** query parameter to the target **Project Code**.



When you later run the parent integration from the Configure and run page, the integration version, integration code, and project code parameters are available.



10. Click **Run**. The child integration in the project is now invoked, instead of the globally-available version.

Information about dynamically invoking a child integration is provided. See [Dynamically Invoke a Child Integration](#).

Manage Accelerator Projects

You can install accelerator projects from the Integration Store. You can extend (customize) integrations in the accelerator project and later upgrade an accelerator project to a newer version automatically without losing any extensions you made to the previous version.

Topics:

- [Extend an Integration in an Accelerator Project](#)
- [Upgrade an Accelerator Project and Merge Extensions](#)

Extend an Integration in an Accelerator Project

You can extend (customize) an integration in an accelerator project by adding and configuring an extension group. An extension group enables you to extend your integration by adding invoke connections; stitch, for-each, switch, map, and integration actions; and global variables to the integrations in your accelerator project.

- [Learn About Accelerators Available in the Integration Store](#)
- [Extend an Integration Before or After an Invoke Connection](#)
- [Add a Stitch Action to an Extended Integration](#)
- [Add an Ad-Hoc Map Action to an Extended Integration](#)
- [Add a For-Each Action to an Extended Integration](#)
- [Add a Switch Action to an Extended Integration](#)
- [Add an Integration Action to an Extended Integration](#)
- [Add an Invoke Connection and Associated Map to an Extended Integration](#)
- [Add a Global Variable to an Extended Integration](#)
- [Add an Extension Group Before a Stage File Action Configured with a Write File Operation](#)
- [Add Connections to an Extended Integration](#)

- Add Lookups and JavaScript Libraries to an Extended Integration
- Merge Extension Group Changes into Updated Accelerator Projects

Learn About Accelerators Available in the Integration Store

The Integration Store provides accelerator projects that you can install and extend in a project. Newer, updated versions of accelerator projects can also become available in the Integration Store for you to re-install, as necessary.

See Find Recipes and Accelerators and Install Recipes and Accelerators in *Getting Started with Oracle Integration 3*.

Extend an Integration Before or After an Invoke Connection

You can extend an integration to add the following before or after an invoke connection:

- Stitch action
- For-each action
- Switch action
- Map action
- Integration action (and associated map) for invoking a child integration
- Invoke connection (and associated map)

1. In the navigation pane, click **Projects**.
2. Find the accelerator project to extend. The **Accelerator** and **Oracle** labels identify these types of projects. You can also use the **Filter** to display a specific project type (all, user-developed, accelerator, extended accelerator, or recipe). For example:



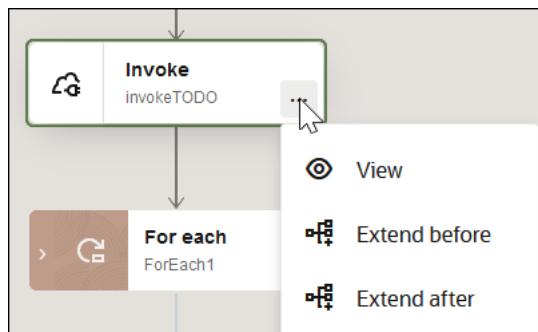
3. Click the project name or click .
4. In the **Integrations** section, find the specific integration to extend.
5. Click **Actions ...**, and select **Extend**. The **Extend** option is only available in accelerator projects. This option behaves similar to the **Edit** option in non-accelerator integrations. The integration is displayed in the canvas.



6. Perform one of the following actions before or after an invoke action.
 - a. Add at the location where you want to add the extension, then select **Extension Group**.

or

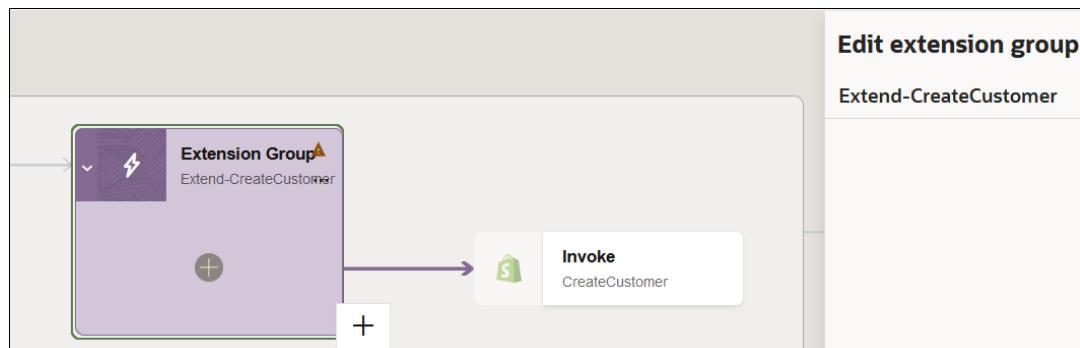
- Click **Actions ...** inside the invoke action.
- Based on where you want to place the extension, select **Extend before** or **Extend after**. Both options are available if you have not yet added an extension to either side of the invoke.



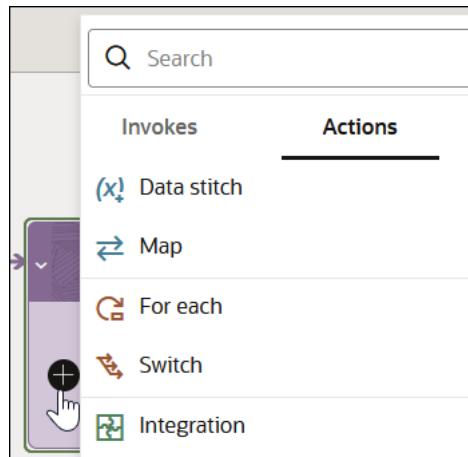
Note:

If **Extend before** and **Extend after** are not available for selection and you have not added an extension to either side of the invoke, your administrator has restricted the use of extensions on the invoke connection. You cannot extend restricted invokes.

The extension group is added to the integration.



- If you want to update the name, click in **Edit extension group**.
- Click **Integration actions** and drag an invoke or action into the integration or click **Add** inside the extension group to add an invoke or action to the integration. For example, if you click **Add** , the following menu is displayed.



See the following sections for details:

- [Add a Stitch Action to an Extended Integration](#)
- [Add an Ad-Hoc Map Action to an Extended Integration](#)
- [Add a For-Each Action to an Extended Integration](#)
- [Add a Switch Action to an Extended Integration](#)
- [Add an Integration Action to an Extended Integration](#)
- [Add an Invoke Connection and Associated Map to an Extended Integration](#)

Add a Stitch Action to an Extended Integration

1. If you select **Actions**, then **Data stitch**:

You can incrementally build a message payload from one or more existing payloads with a stitch action. For example, a data stitch enables you to customize the sales order created in an inbound application. Assume you have a use case in which whenever a sales order is created in an inbound application, your integration must retrieve and push information in a custom object to an outbound application. In the inbound application, the custom object consists of the following attributes.

Sales	* OrderManager [®] Name	Account	Item Description	Item Name	Order Status	Quantity Purchased	Registry ID	Requested Ship Date
	Order 97663 - It					2		m/d/y
Quotes and Ord...	Order 97664 - It		Vario 6500 Table	AS46337	Processing	4		1/24/2
Leads	Order 97665 - It					1		m/d/y
Relationships	Order 97666 - It					1		m/d/y
Service Requests	Order 97667 - It		Vario 6500 Table	AS46337	Processing	2		m/d/y
Notes	Order 97668 - It					4		m/d/y
Assessments	Order 97669 - It		Vario 6500 Table	AS46337	:synced	5		m/d/y
Activities	Order 97670 - It				Processing:-syn	2		m/d/y
Interactions Hist...	Order 98663 - It		Vario 6500 Table	AS46337	Processing:-syn	3		m/d/y
OrderManage...								

The Configure stitch panel opens.

- Configure the stitch. This action enables you to customize the integration with mappings to or from attributes of custom objects. For this example, a stitch variable is created to process the order status attribute in the custom object.

The screenshot shows a 'Configure stitch' dialog box. At the top, it says 'Stitch1'. Below that, a note says 'Specify as many variables for this stitch as you need, and configure their values.' Underneath, there's a text input field containing the assignment statement: 'OrderStatus_c' = 'concat(OrderStatus_c, ':', '-synced')'. The 'OrderStatus_c' part is highlighted with a green background.

See [Build Complex Assignment Statements with a Stitch Action](#).

- When integration design is complete, click **Save**. Assume you later need to further customize the custom objects in the inbound application that triggers your integration. For example, an additional field to handle the **Item Number** is added to the custom object.

This use case is not about installing a newer version of the accelerator project available from the Integration Store. Rather, this is an informal user update to the custom objects. You can handle this use case by refreshing the endpoints to pick up the new object and editing the data stitch action to process the new object. You must use a new identifier and/or version. You can also set the name, keywords, and description.

	Item Description	Item Name	Order Status	Quantity Purchased	Registry ID	Requested Ship Date	Unit List Price	Item Number
Sales				2		m/d/y	150	
Quotes and Ord...								
Leads	Vario 6500 Table	AS46337	Processing	4		1/24/2	150	
Relationships				1		m/d/y	150	
Service Requests				1		m/d/y	150	
Notes	Vario 6500 Table	AS46337	Processing	2		m/d/y	150	
Assessments			:synced	4		m/d/y	150	
Activities	Vario 6500 Table	AS46337	Processing:-syn	5		m/d/y	150	
Interactions Hist...	Vario 6500 Table	AS46337	Processing:-syn	2		m/d/y	150	-999
OrderManage...	Vario 6500 Table	AS46337	Processing:-syn	3		m/d/y	150	-999

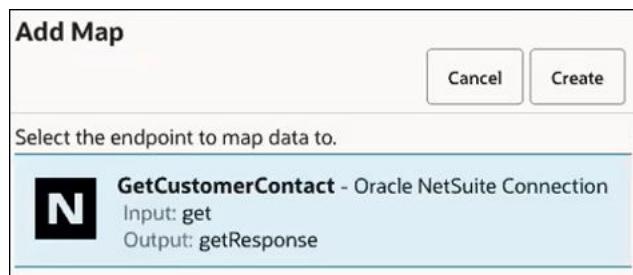
- Click **Actions ...**, then select **Refresh Endpoints**. This action fetches the latest attributes added to the custom object.
- Open the stitch action in the extension group and add the latest custom attribute mapping to the extension group. For this example, a second stitch variable is created to process the item number attribute in the custom object.



Add an Ad-Hoc Map Action to an Extended Integration

- If you select **Actions**, then **Map**:

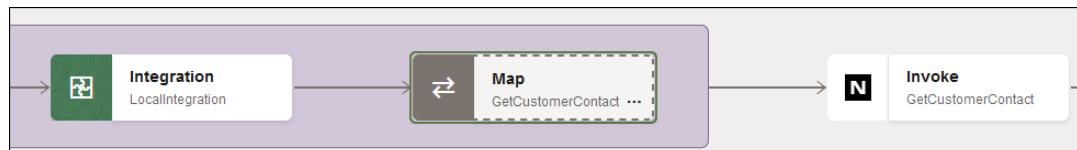
The Add Map panel opens to show the endpoints available to which to map. Endpoints both inside and outside the extension group are shown.



If there are no endpoints available to which to map, the following message is displayed:

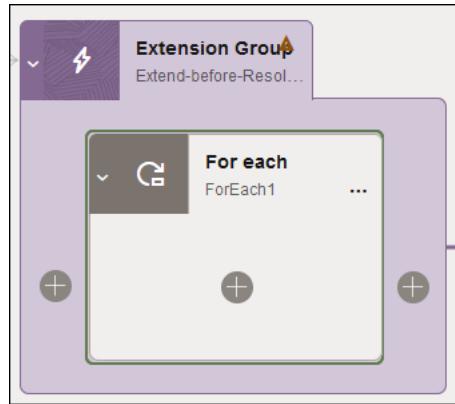
No outputs available to map to

- Select the endpoint, and click **Create**.
The mapper opens.
- Map appropriate source elements to target elements.
A map action is visible in the **Extension Group**.



Add a For-Each Action to an Extended Integration

- If you select **Actions**, then **For each**:
A for-each action is added to the **Extension Group**.



- Configure the for-each action. For this example, an **Account** element is defined as the repeating action.

Input sources

Sources

- process *
- Account
- \$self
- \$tracking_var_1
- \$tracking_var_2
- \$tracking_var_3

: Configure for each

ForEach1

Repeating element * Account

To update this value, drag and drop from source tree.

Current element name * f0_Account

- Perform further configuration, as necessary. For this example, a stitch action is added. For each iteration of the **Account** element, a value is updated to the current time.

Input sources

Components

Functions

- Advanced
- Boolean
- Conversion
- Date
 - adjust-date-to-timezone
 - adjust-dateTime-to-timezone
 - adjust-time-to-timezone

: Configure stitch

Stitch1

Specify as many variables for this stitch as you need, and configure values.

UpdatedTime = current-time()

Variable UpdatedTime

= Operation Assign

Value current-time()

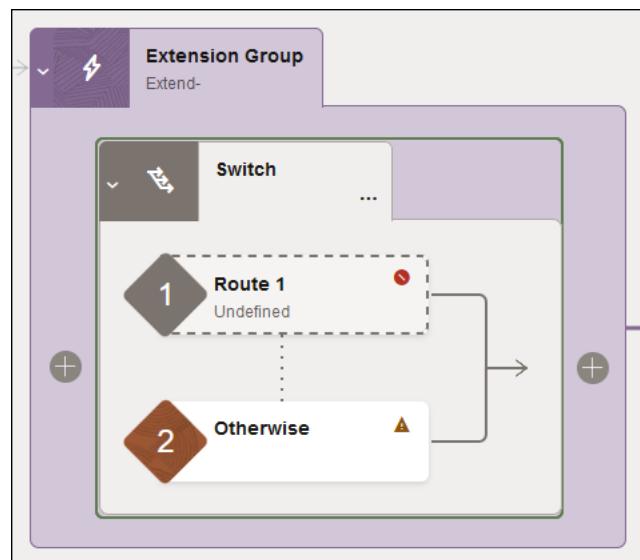
The extension group design looks as follows:



Add a Switch Action to an Extended Integration

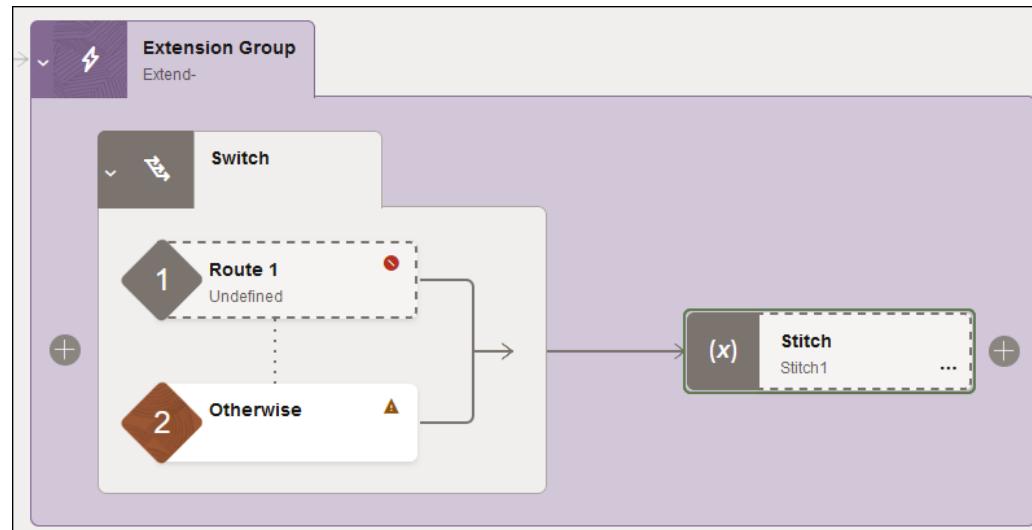
1. If you select **Actions**, then **Switch**:
A switch action is added to the **Extension Group**.

You can add routing expressions in an integration with a switch action.



You can add a stitch action inside the switch action.

- a. Click **+**, then select **Stitch**.



- b. Configure the stitch action. For example:

Configure stitch

Stitch1

Specify as many variables for this stitch as you need, and configure their values.

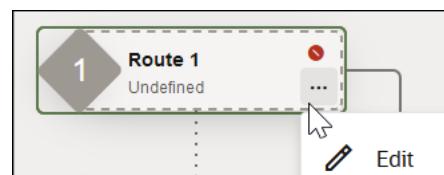
tracking_var_2 = tracking_var_3

Variable tracking_var_2

= Operation Assign

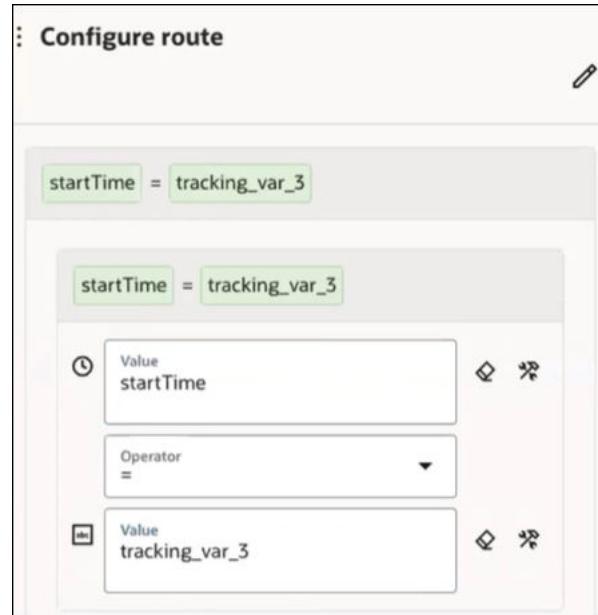
Value tracking_var_3

- c. Inside **Route 1**, click **Actions ...**, then select **Edit**.

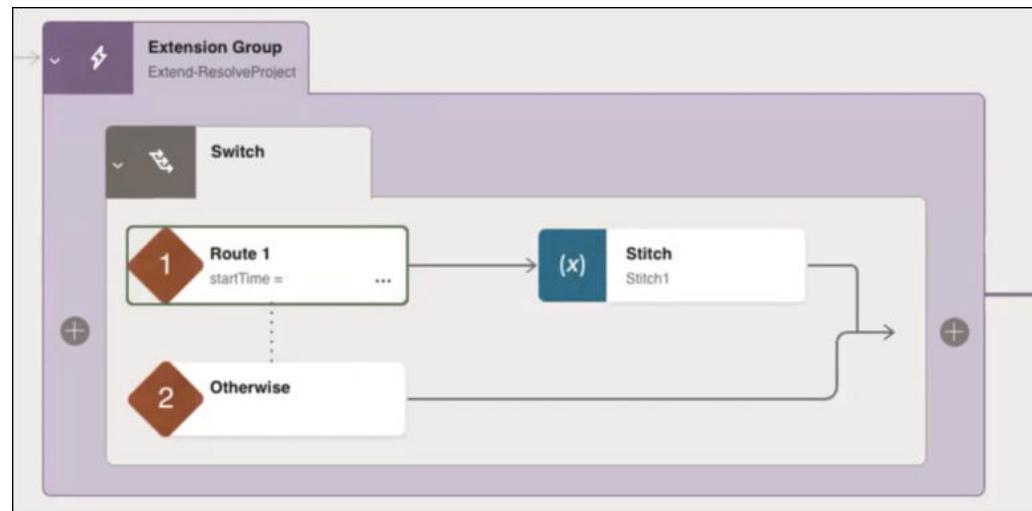


The Configure route panel opens.

- d. Configure the switch action. For example:

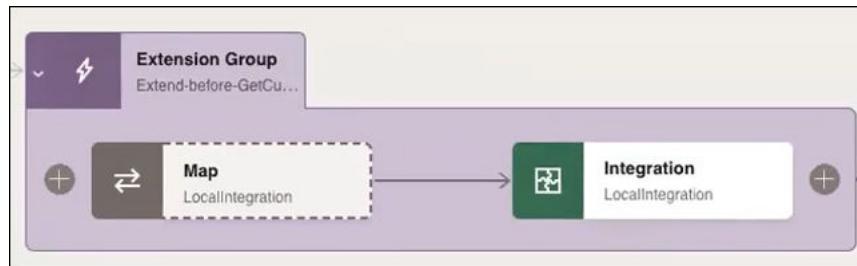


The extended portion of the integration looks as follows.



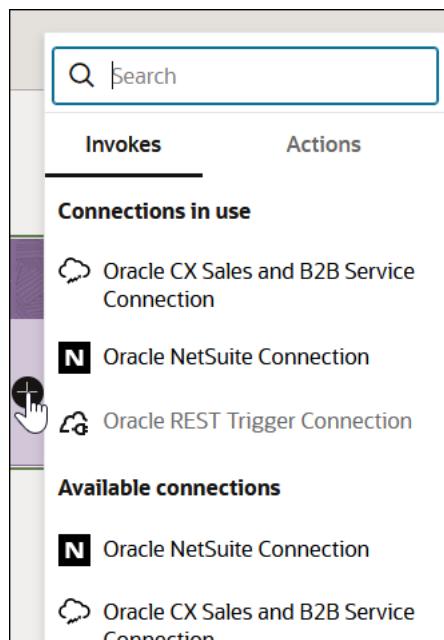
Add an Integration Action to an Extended Integration

1. If you select **Actions**, then **Integration**:
The Integration Adapter Wizard is displayed.
2. Click through the pages of the wizard to select the child integration to invoke.
An integration action and associated map action are added to the **Extension Group**.



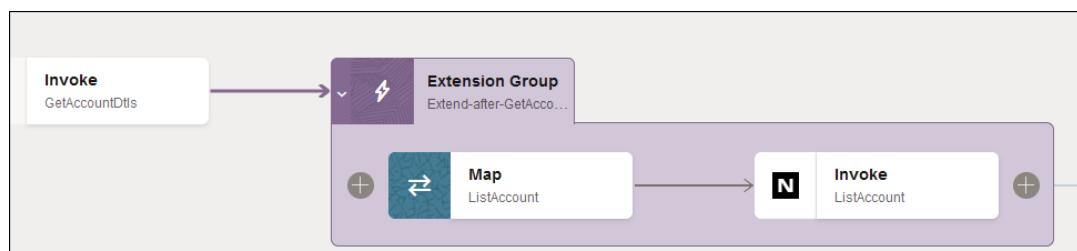
Add an Invoke Connection and Associated Map to an Extended Integration

- If you select **Invokes**, then select an invoke adapter connection:



The Adapter Endpoint Configuration Wizard is displayed for the selected invoke connection.

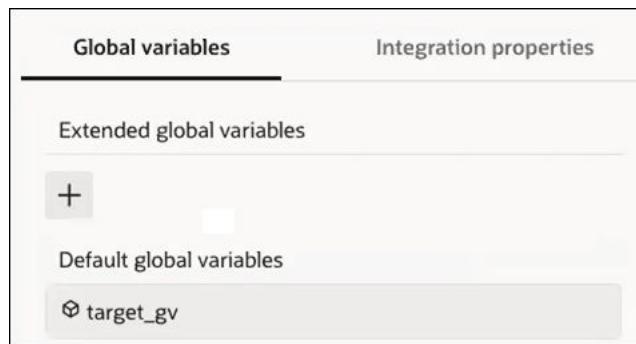
- Click through the pages of the wizard and configure the invoke connection.
- When complete, the configured invoke connection and an associated map action are added to the **Extension Group**.



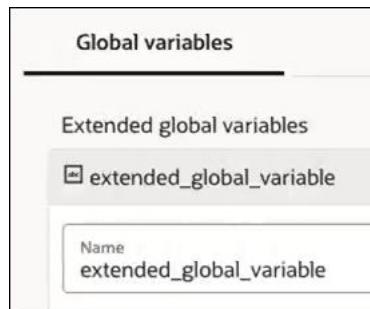
- Double-click the map action and map appropriate source elements to target elements.

Add a Global Variable to an Extended Integration

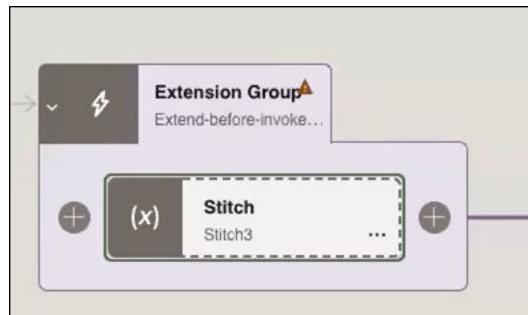
1. In the **Integrations** section, find the specific integration to extend.
2. Click **Actions ...**, and select **Extend**.
The integration is displayed in the canvas.
3. In the right pane, click **Global Variables** .
There are two types of global variables:
 - **Extended global variables**: Variables that you can create to extend the accelerator for your business requirements.
 - **Default global variables**: Variables that are defined by the producer of the extended accelerator. You cannot edit these variables.



4. Click  to add your own extended global variable. For this example, `extended_global_variable` is added.



5. Add an **Extension Group** to the integration.
6. Add an action in which to use the extended global variable. For this example, a stitch action is added.



7. Configure the stitch action to use the extended global variable.

Input sources	Functions
Sources ▾ \$triggerTODO ▶ execute * ▶ \$invokeTODO ▶ \$invokeTODO_REQUEST ▶ \$self ▶ \$target_gv ▶ \$triggerTODO_RESPONSE (x) \$extended_global_variable (x) \$tracking_var_1 (x) \$tracking_var_2	: Configure stitch Stitch3 <small>Specify as many variables for this stitch as you need, and configure their values.</small> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> extended_global_variable = tracking_var_1 </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> (x) Variable extended_global_variable </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> = Operation Assign </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> (x) Value tracking_var_1 </div>

Note:

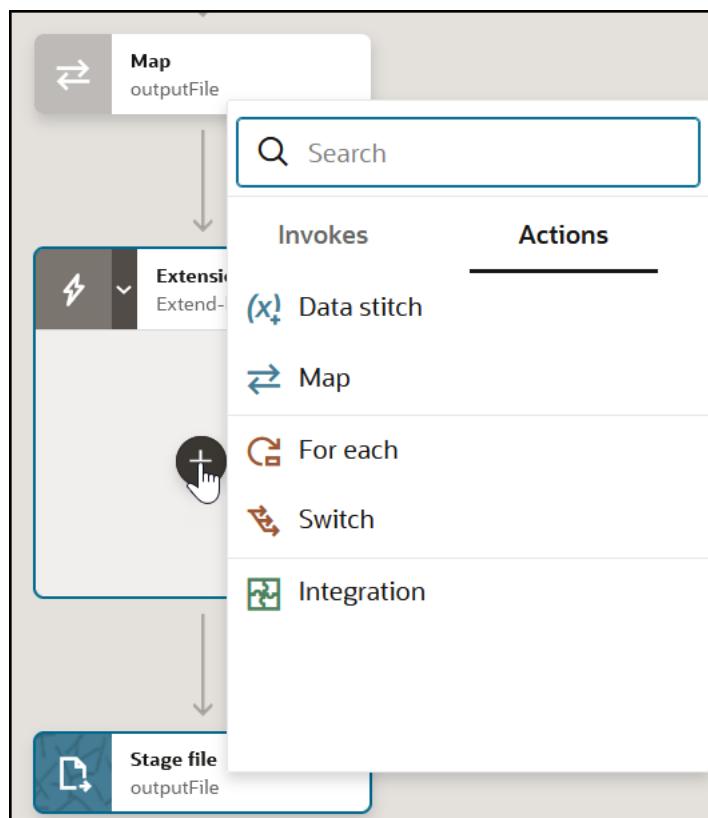
If there is a naming conflict with global variables (for example, two global variables have the same name), the entire merge is canceled. This cancellation occurs because a global variable can be used in any extension group. For example, if you create an extended global variable named **conflict_gv**, and a default global variable of the same name already exists, the merge is canceled, and an error is displayed.

- **Merge is aborted, source and target integrations have naming conflicts in global variables: conflict_gv.**
- **Extended Global Variable: conflict_gv**
 Target: gtv1
 Extended global variable in source.

Add an Extension Group Before a Stage File Action Configured with a Write File Operation

You can add an extension group before a stage file action that is configured to write to a file. This is the only stage file action operation to which you can add an extension. This operation includes a preceding map action. For example, you may want to extend the contents of the map action with a data stitch action.

1. Perform one of the following actions before the stage file action.
 - a. Add at the location where you want to add the extension, then select **Extension Group**.
 - or
 - a. Click **Actions ...** inside the stage file action and select **Extend before**.
2. Add the appropriate action (for example, a data stitch action).



Add Connections to an Extended Integration

1. In the navigation pane, click **Projects**.
2. Find the accelerator project to extend.
3. In the **Connections** section, click **Add** .
4. Select an adapter and click **Create**.
5. Enter a name. Note that the prefix **BAEXT_** is automatically added to the name in the **Identifier** field to indicate that this is an extended connection.

Create connection

 REST adapter

Name
RestConn
Enter a name using letters (A-Z,a-z), numbers (0-9), spaces () and special characters (_ -). It cannot be longer than 50 characters and must start with a letter.

Identifier
BAEXT_RESTCONN

Role
Trigger ▾

Keywords

Description

Share with other projects

6. Click **Create**, then define your connection properties and security. Once complete, the extended connection appears in the **Connections** section, and can be used in integrations.

Add Lookups and JavaScript Libraries to an Extended Integration

To add a lookup to an extended integration.

1. In the navigation pane, click **Projects**.
2. Find the accelerator project to extend.

3. In the **Lookups** section, click **Add** , then click **Create**.
Note that the prefix `BAEXT_` appears by default in the **Name** field. An extended lookup name must be prefixed with this value. If you delete the `BAEXT_` prefix and attempt to create the lookup, you receive a warning.

The screenshot shows a 'Create lookup' dialog box. At the top left is the title 'Create lookup' and a help icon. Below the title is a placeholder text 'Enter details'. There are three input fields: 'Name' (containing 'BAEXT_'), 'Keywords', and 'Description'.

4. Add a name after the prefix `BAEXT_`.
5. Enter or select an existing keyword, and an optional description.
6. Click **Create**.
7. Design your lookup.
Once complete, the extended lookup appears in the **Lookups** section, and can be used in integrations.

To add a JavaScript library to an extended integration.

1. In the navigation pane, click **Projects**.
2. Find the accelerator project to extend.
3. In the **Libraries** section, click **Add**
4. Drag a JavaScript library file into the Import panel.
The **Name**, **Identifier**, and **Version** fields are prepopulated with details from the file you imported. Note that the prefix `BAEXT_` is automatically added to the function name in the **Identifier** field to indicate that this is an extended JavaScript library.

The screenshot shows a 'Import' dialog box. At the top, it says 'Upload a ZIP, JAR, JS file'. Below that, a file named 'addfunction.js' (0.10 KB) is listed with a delete icon. There are four input fields: 'Name' (addfunction), 'Identifier' (BAEXT_ADDFUNCTION), 'Version' (01.00.0000), and 'Description' (empty). A question mark icon is in the top right corner.

5. Click **Import**.
6. Complete design of your JavaScript library.

Once complete, the extended JavaScript library appears in the **Libraries** section, and can be used in integrations.

Merge Extension Group Changes into Updated Accelerator Projects

Oracle can periodically update and upload newer versions of accelerator projects to the Integration Store for your consumption. You can upgrade an accelerator project to this newer version automatically without making manual changes to the existing installation. During upgrade, you are prompted to automatically merge any customizations you performed in the extension group of the previous version into the newer version of the accelerator project. See [Upgrade an Accelerator Project and Merge Extensions](#).

on an invoke connection also includes options for **Extend before** or **Extend after**. This enables you to install the supported accelerator actions before or after the invoke connection.

Upgrade an Accelerator Project and Merge Extensions

You can install and then extend (customize) accelerator projects in your instance. Oracle can periodically update and upload newer versions of these accelerator projects to the Integration Store for your consumption. You can upgrade an accelerator project to this newer version automatically without losing any extensions you made to the existing installation.

- [Understand the Rules for Merging Extensions](#)
- [Automatically Merge Extensions](#)
- [Manually Merge Extensions](#)

Understand the Rules for Merging Extensions

Before upgrading an extended accelerator project, it is important to understand the rules for merging extensions.

Rule	Example
When you have multiple extended (customized) versions of an accelerator project installed, and select to install a newer version, the installer checks for extensions in the previously installed highest version. If that version includes extensions, they are automatically merged into the newer version.	You currently have the following accelerator project versions installed: <ul style="list-style-type: none">• Versions 2.0.0 (which includes extensions)• Versions 4.0.0 (which includes extensions) If you install version 6.0.0, the extensions made in version 4.0.0 (previously installed highest version) are automatically merged into version 6.0.0. The extensions in version 2.0.0 are ignored.
When you have multiple versions of an accelerator project installed, and select to install a newer version, the installer checks for extensions in the previously installed highest version. If that version does <i>not</i> include extensions, nothing is merged into the newer version.	You currently have the following accelerator project versions installed: <ul style="list-style-type: none">• Version 2.0.0 (which includes extensions)• Version 4.0.0 (which does <i>not</i> include extensions) If you install version 6.0.0, the installer checks for extensions in the previously installed highest version. In this case, no extensions are applied because version 4.0.0 included no extensions. The extensions in version 2.0.0 are ignored.
When you have a single version of an accelerator project installed, and select to install a newer version, the installer checks for extensions in the previously installed highest version. If that version includes extensions, they are automatically merged into the newest version.	You currently have the following accelerator project version installed: <ul style="list-style-type: none">• Version 1.0.0 (which includes extensions) If you install a higher version of the accelerator project (for example, version 2.0.0, 3.0.0, 4.0.0, or higher), the extensions are applied to the newly installed version.
When you have multiple extended versions of an accelerator project installed, and select to install an <i>older</i> version, the installer does <i>not</i> check for extensions because you are installing a lower version.	You currently have the following accelerator project versions installed: <ul style="list-style-type: none">• Versions 2.0.0 (which includes extensions)• Versions 6.0.0 (which includes extensions) If you install version 4.0.0, the extensions made in version 6.0.0 are <i>not</i> merged into version 4.0.0 because you are installing a lower version. The extensions in version 2.0.0 are ignored.

Automatically Merge Extensions

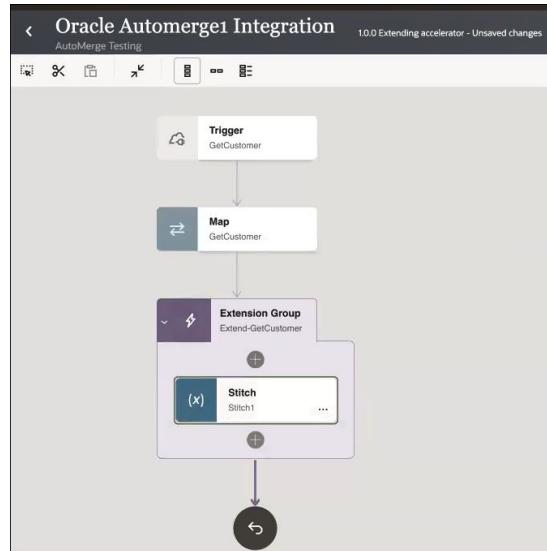
This section provides an overview of how to upgrade to a newer extended accelerator project and automatically merge your extensions (customizations). This process automatically merges the extensions in all integrations in the project.

Assume you have an accelerator project that includes two integrations on version 1.0.0.

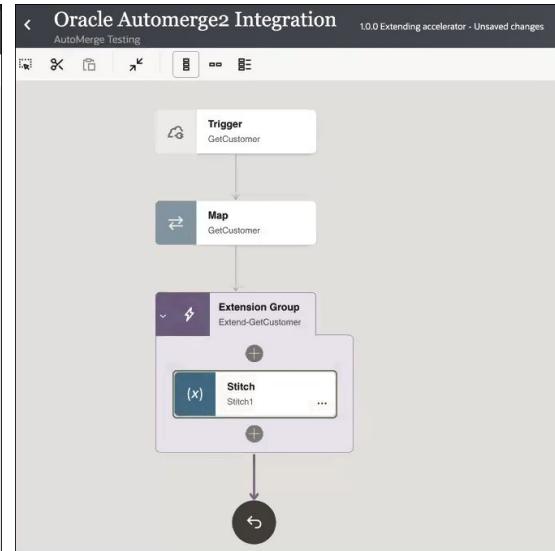
The screenshot shows the Oracle Integration AutoMerge Testing interface. At the top, there's a navigation bar with a back arrow, the title "AutoMerge Testing", and tabs for "Design" and "Deploy". Below the title, there's a search bar with the placeholder "Full or partial name or keyword or description". On the left, under "Details", there are sections for "Description" (empty), "Keywords" (empty), "Created by" (l2serviceadmind), and "Created on" (Aug 9, 2023, 07:08:16 PM IST). On the right, under "Integrations", there are two entries: "Oracle Automerge1 Integrat..." version 1.0.0 and "Oracle Automerge2 Integrat..." version 1.0.0, both marked as "Configured".

Both integrations include an extension group with a stitch action that you added.

Oracle Automerge1 Integration

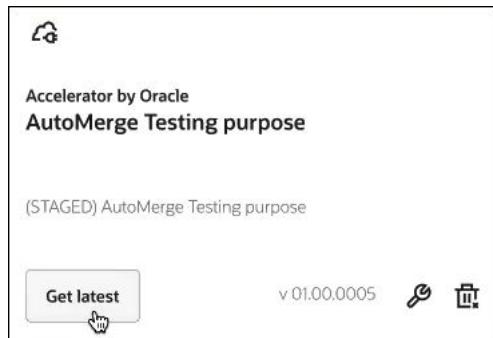


Oracle Automerge2 Integration



Newer versions of this accelerator project are then uploaded to the Integration Store. You want to install a newer version and not lose the extensions you made to your current version (1.0.0).

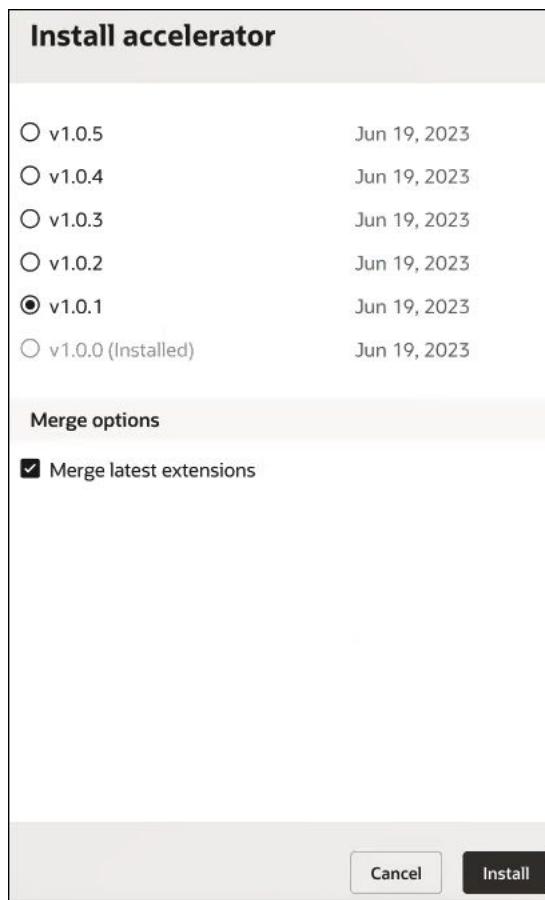
1. On the Oracle Integration Home page, scroll to the **Accelerators & Recipes** section.
2. Find the accelerator project to upgrade. The **Get latest** label indicates that a new version of the accelerator project is available for installation.
3. Click **Get latest**.



You can also configure and delete installations from this location.

The Install accelerator panel opens and shows the latest accelerator versions to which you can upgrade.

4. Select the version to which to upgrade (for this example, **v1.0.1** is selected).
5. If you want to merge the extensions you added (for this example, the stitch actions you added to both version 1.0.0 integrations), then click **Merge latest extensions**.
6. Click **Install**.

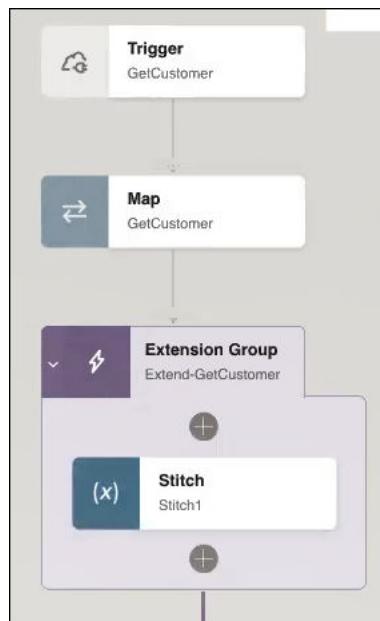


7. Click the link in the Confirmation message that is displayed. This takes you to your project to view installation progress.
8. Click **Refresh**  periodically.

9. Note that the latest installed versions (1.0.1) are successfully merged above the initial integration versions (1.0.0).

Integrations				
	→	Oracle Automerge1 Integration	1.0.1	Merge successful
	→	Oracle Automerge2 Integrati...	1.0.1	Merge successful
	→	Oracle Automerge1 Integrat...	1.0.0	Configured
	→	Oracle Automerge2 Integrat...	1.0.0	Configured
View all 4 integration(s)				

10. Click the latest integrations (1.0.1) to see that your customizations have been merged.



11. Click **Merge report** to view a report about the merge.

Extension merge report

Merge date: Aug 9th, 2023 07:11:04 PM IST Merged by: l2serviceadmin

Source version: 1.0.0 Target version: 1.0.1

Error(s): 0 Warning(s): 0

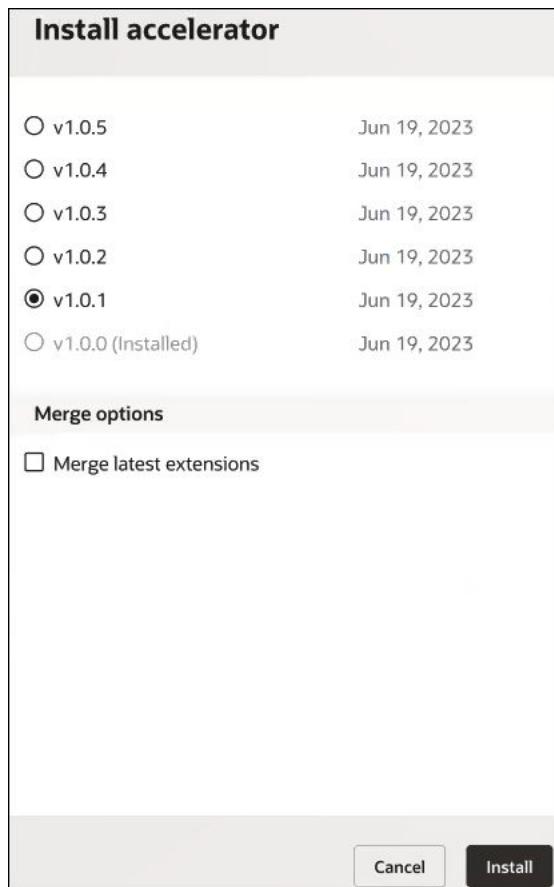
- **Extension Group: Extend-GetCustomer**
Target: Extend-GetCustomer
Source extension group.
- **Extension Group: Extend-GetCustomer**
Target: Extend-GetCustomer
Extension group passed extension group check.
- **Extension Group: Extend-GetCustomer**
Target: Extend-GetCustomer
Merged try element.
- **Extension Group: Stitch1**
Target: sh0
Merged stitch element.

12. Exit the integration.
13. Click **Save** when prompted.

Manually Merge Extensions

If you do not select the **Merge latest extensions** check box described in [Step 5](#), your customizations are not applied. However, you still have the opportunity to manually merge your extensions.

1. Select the version to install, but do not click **Merge latest extensions**.



2. Go to the project.
3. In the **Integrations** section, click the newly installed version. Note that the status label is listed as **Configured**, and not **Merge Successful** for the two newly-installed integrations.

Integrations				
	Oracle Aut... 1.0.1	Configured	...	
	Oracle Automerge1 Integration 1.0.1	Configured	...	
	Oracle Automerge1 Integrat... 1.0.0	Configured	...	
	Oracle Automerge2 Integrat... 1.0.0	Configured	...	

The Extended accelerator versions dialog prompts you to select to merge your extensions.

Version	Last updated	Status
1.0.0	Aug 9, 2023 07:09:04 PM IST	Configured

Cancel Skip Select

4. Click **Select** to merge your extensions. You can also click **Skip** if you do not want to merge your extensions.
5. Repeat these steps for any remaining integrations that include extensions you want to manually merge.

Activate or Deactivate an Integration in a Project

You can individually activate or deactivate an integration in a project.

To define a local invoke, the target integration must be activated.

Activate an Integration

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, find the specific integration to activate.
4. Click **Actions ...**, and select **Activate**.
The Activate integration panel opens.
5. Follow the steps to activate an integration. For REST Adapter-triggered and AS2 Adapter-triggered integrations, a selection is available for replaying the integration during runtime on the Instances page. See [Activate and Deactivate Integrations](#) and [Replay Integration Instances](#).

Deactivate an Integration

Assets can be deactivated individually regardless of whether they were activated individually or as part of a project.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, find the specific integration to deactivate. Activated integrations are identified by the **Active** label.
4. Click **Actions ...**, and select **Deactivate**. See [Activate and Deactivate Integrations](#).

Deploy Integration Endpoints to Oracle Cloud Infrastructure API Gateway

You can deploy individual integration endpoints as routes to Oracle Cloud Infrastructure API Gateway. An Oracle Cloud Infrastructure API Gateway instance supports a maximum of 20 deployments. Each deployment can contain up to 50 routes (each routing to individual endpoints). This provides you with a capacity of 1000 integration endpoints to which to deploy.

See [Overview of API Gateway](#).

- [Restrictions](#)
- [Create a Dynamic Group and Policy to Grant Gateway Access](#)
- [Create a Virtual Cloud Network and an Internet Gateway](#)
- [Create an Oracle Cloud Infrastructure API Gateway and Deployment in Oracle Cloud Infrastructure Console](#)
- [Deploy an Integration to API Gateway](#)

 **Note:**

To perform tasks in the Oracle Cloud Infrastructure Console, you must have the ServiceAdministrator role.

Restrictions

- Deployment to Oracle Cloud Infrastructure API Gateway only works in identity domain-enabled environments.
- Only integrations in a project can be deployed to Oracle Cloud Infrastructure API Gateway. The integration must be publicly available. That is, the **Available to other projects** check box must be selected for this integration.
- The integration must be activated.
- You can only deploy a REST Adapter trigger-based integration.
- REST Adapter trigger connections that expose multiple entry points to a single integration are *not* supported. See [Receive Requests for Multiple Resources in a Single REST Adapter Trigger with a Pick Action](#).

See [API Limits](#).

Create a Dynamic Group and Policy to Grant Gateway Access

To deploy integration endpoints to Oracle Cloud Infrastructure API Gateway, you must satisfy the following prerequisites in the Oracle Cloud Infrastructure Console.

- Create a dynamic group.
- Create a policy to grant access to Oracle Cloud Infrastructure API Gateway.

You create the required dynamic group and assign a policy to that group to allow your Oracle Integration instance to access Oracle Cloud Infrastructure API Gateway. The policy defines the permissions for the dynamic group and determines which operations the dynamic group can perform in the Oracle Cloud Infrastructure API Gateway.

1. Log in to the Oracle Cloud Infrastructure Console.
2. Obtain the client ID of the OAuth application for the Oracle Integration instance.

3. In the upper right corner, select **Profile**, then click the identity domain.
4. In the left navigation pane, click **Oracle Cloud Services**.
The Oracle Cloud Services page for your domain appears.
5. In the **Name** column, click your service instance.
6. Scroll down to the **General Information** section and copy the client ID value to use to create your dynamic group.
7. Scroll to the breadcrumbs at the top and click **Default domain**.



8. In the left navigation pane, click **Dynamic groups**.
9. Click **Create Dynamic Group**.
10. Enter the following details:
 - a. In the **Name** and **Description** fields, enter values. These fields are required.
 - b. In the **Matching Rules** section, enter the required rule. The resource ID you specify must match the client ID of the OAuth application of your Oracle Integration instance. Ensure that you enclose the value in single quotes. For example:

```
resource.id = 'client_ID'
```

A screenshot of the 'Create dynamic group' dialog box. It has a title bar 'Create dynamic group'. Below it, there's an info icon with a note about matching rules. Two radio buttons are shown: 'Match any rules defined below' (selected) and 'Match all rules defined below'. A 'Rule 1' section contains a text input field with the value 'resource.id = 'F2AC34958399A1G''. There's also a 'Rule builder' link and a close button 'X'.

11. Scroll to the breadcrumbs at the top and click **Identity**.



12. In the left navigation pane, click **Policies**.

13. Click **Create Policy**.
14. Select the compartment in which to create the policy.
15. Enter the following details:
 - a. In the **Name** and **Description** fields, enter values. These fields are required.
 - b. In the **Policy Builder** section, build the required policy for the dynamic group. For example:

```
allow dynamic-group dynamic_group to manage api-gateway-family in
compartment compartment_name
```

Where:

- *dynamic_group*: Is the dynamic group name you created.
- *compartment_name*: Is the compartment in which your Oracle Integration instance is located.

This enables the Oracle Integration instance associated with the dynamic group to call Oracle Cloud Infrastructure API Gateway in this particular compartment.

Create a Virtual Cloud Network and an Internet Gateway

You must create a virtual cloud network and internet gateway before you can create an Oracle Cloud Infrastructure API Gateway.

1. In the navigation menu, go to **Networking**.
2. Click **Virtual cloud networks**.
3. Click **Create VCN**.
4. Enter the following information, then click **Create VCN**.

Element	Description
Name	Enter a virtual cloud network name.
Compartment	Displays the compartment you previously selected.
IPv4 CIDR Blocks	Assign up to five IPv4 CIDR blocks to a VCN. At least one is required. See VCN and Subnet Management .
Use DNS hostnames in this VCN	If you plan to use VCN DNS or a third-party DNS, this is required for instance hostname assignment. This selection cannot be changed after the VCN is created. See DNS in Your Virtual Cloud Network
DNS Label	This value is generated from the virtual cloud network name if not specified.
DNS Domain Name	This value is generated from the virtual cloud network name if not specified.

The details page for the virtual cloud network is displayed.

5. Click **Create Subnet**.
6. Enter the following information, then click **Create Subnet**.

Element	Description
Name	Enter a subnet name.
Compartment	Displays the compartment you previously selected.
Subnet Type	Select a subnet type: <ul style="list-style-type: none">• Regional (Recommended)• Availability Domain-specific
IPv4 CIDR Block	Enter the IPv4 CIDR block.
Route Table Compartment	Select the route table compartment.
Subnet Access	Select an access type: <ul style="list-style-type: none">• Public Access• Private Access
Use DNS hostnames in this VCN	If you plan to use VCN DNS or a third-party DNS, this is required for instance hostname assignment. This selection cannot be changed after the VCN is created.
DNS Label	This value is generated from the virtual cloud network name if not specified.
DNS Domain Name	This value is generated from the virtual cloud network name if not specified.
Dhcp Options Compartment	Select the default DHCP options.
Select Security List Compartment	Select the security list.
Resource Logging	Select to enable resource logging.

7. Under **Resources**, select **Internet Gateways**.
8. Click **Create Internet Gateway**.
9. Enter the following information, then click **Create Internet Gateway**.

Element	Description
Name	Enter an internet gateway name.
Compartment	Displays the compartment you previously selected.

10. Under **Resources**, select **Security Lists**.
11. In the **Name** column, click the default security list.
12. Click **Add Ingress Rules**.
 - a. Specify the source CIDR value.
 - b. Leave the **Source Port Range** field blank.
 - c. Enter 443 in the **Destination Port Range** field.
 - d. Click **Add Ingress Rules**.
13. Return to the details page for the virtual cloud network you created.
14. In the **Resources** section, click **Route Tables**.
15. In the **Name** column of the **Route Tables** section, click the default route rule.
16. Click **Add Route Rules**.
17. Enter the following information, then click **Add Route Rules**.

Element	Description
Target Type	Select Internet Gateway .
Destination CIDR Block	Enter the destination CIDR block.
Target Internet Gateway	Select the API gateway.
Description	Enter an optional description.

18. In the **Resources** section, click **Network Security Groups**.
19. Click **Network Security Group**.
 - a. Enter a name.
 - b. Leave the compartment as is.
20. Click **Next**.
 - a. Select **CIDR** in the **Source Type** list.
 - b. Select **TCP** in the **IP Protocol** list.
 - c. Specify **All** in the **Source Port Range** field.
 - d. Specify the value in the **Source CIDR** field.
 - e. Specify **443** in the **Destination Port Range** field.
 - f. Click **Create**.
21. In the breadcrumbs at the top of the page, click the link for the virtual cloud network that you created.
22. In the **Resources** section, click **Security Lists**.
If egress rules do not exist, you must define them.
23. In the **Resources** section, click **Egress Rules**.

Create an Oracle Cloud Infrastructure API Gateway and Deployment in Oracle Cloud Infrastructure Console

You must create an Oracle Cloud Infrastructure API Gateway. Each gateway instance supports a maximum of 20 deployments. Each deployment can handle up to 50 routes. This means that one gateway instance can protect up to 1000 APIs. You can select an existing deployment, enabling a new route to be created in that deployment. You can also create a new deployment.

1. In the navigation menu, go to **Developer Services**.
2. Under **API Management**, select **Gateways**.
3. Select the compartment to use for deployment.
4. Click **Create Gateway**.
5. Enter the following information, then click **Create Gateway**.

Element	Description
Name	Enter a gateway name.
Type	Select Public .
Compartment	Displays the compartment you previously selected.

Element	Description
Network	Select the following networking details: <ul style="list-style-type: none"> Virtual cloud network: Select a virtual cloud network. Subnet: Select a VCN with at least one regional subnet added. See Networking Overview .
Enable network security groups	Select the check box, then select a compartment with at least one network security group.
Certificate	Select an SSL/TLS certificate that has been added to Oracle Cloud for use with a custom DNS configuration or use the default certificate provided by the gateway. See Setting Up Custom Domains and TLS Certificates .

The new gateway is displayed in the **Name** column on the Gateways page.

The screenshot shows the 'Gateways' page in the Oracle Cloud API Gateway interface. The title bar says 'Gateways in development-I2 compartment'. On the left, there's a sidebar with 'Gateways' selected, and other options like 'APIs', 'Certificates', 'Usage plans', and 'Subscribers'. The main area has a 'Create Gateway' button. A table lists the gateway: Name: test-oic-apigw, State: Active, Type: Public, Created: Tue, Nov 7, 2023, 09:05:26 UTC, Updated: Tue, Nov 14, 2023, 07:41:34 UTC. At the bottom right of the table, it says 'Showing 1 item < Page 1 >'.

6. Click the gateway name.
7. Under **Resources**, click **Deployments**.
8. Click **Create deployment**.

The screenshot shows the details page for the 'test-oic-apigw' gateway. At the top, there are buttons for 'Edit', 'Move resource', 'Add tags', and 'Delete'. Below that, there are tabs for 'Gateway information' (selected) and 'Tags'. Under 'Gateway information', fields include OCID, Compartment, Type (Public), Subnet, Hostname, IP address, and creation/updated dates. Below this is a 'Deployments' section with a 'Create deployment' button and a table for managing deployments.

Name	Path prefix	State	Endpoint	Deployed

The Create deployment wizard is displayed.

You can create deployments based on categories appropriate to your business environment. For example, you may want to create separate deployments for applications, functional areas within an application, client requirements (for example, all APIs for a client-facing portal), and so on. You then deploy the integration endpoints to the appropriate deployment category.

9. Enter the following information, then click **Next**.

Element	Description
Name	Enter a deployment name. For this example, Netsuite is entered.
Path prefix	Enter a prefix. For this example, /netsuite is entered.
Compartment	Displays the compartment you previously selected.
API request policies	Select API request policies as required for your environment: <ul style="list-style-type: none"> Mutual-TLS: Select to enable mTLS. CORS: Configure CORS access. Rate limiting: Configure rate limiting. Usage plans: Configure usage plans.
API logging policies	Select a logging level. See Adding Logging to API Deployments .
Tags (under Show advanced options)	Add tags to organize your resources.

- 10.** Select an **Authentication** method, and specify additional details, then click **Next**
- **No Authentication:** Any client that has network access to the gateway can make requests to all routes in this deployment.
 - **Single Authentication:** Configure integration with a single identity provider. You can optionally limit access for all routes to authenticated clients only.
If you select this option, authentication, validation, and other security fields are displayed for configuration.
 - **Multi-Authentication:** Configure integration with one or more identity providers. You can optionally limit access for all routes to authenticated clients only.
If you select this option, additional authentication fields are displayed for configuration.

Route 1 is displayed.

Individual integration endpoints are deployed as routes to an Oracle Cloud Infrastructure API Gateway deployment.

- 11.** Enter the following information, then click **Next**.

Element	Description
Path	Enter a path (for example, /order).
Methods	Select one or more methods based upon your requirements (for example, GET , POST , PUT , or others).
Add a single backend	This option enables all requests for this route to be sent to the same backend. Select a backend type. Each selection causes additional fields to be displayed for you to configure. <ul style="list-style-type: none"> HTTP Oracle functions Stock response Logout

Element	Description
Add multiple backends	<p>Route to different backends based on criteria at runtime.</p> <p>To configure your route to support multiple backends, first define the request context element to use as the selector. The gateway uses the selector at runtime to choose the backend based on a matching rule defined for the backend.</p> <p>Selector: Select the request context table. Some selections cause additional fields to be displayed for you to configure.</p> <ul style="list-style-type: none"> Auth Headers Host Path Query parameters Subdomain Usage plan ocid <p>Backends: Add one or more backends for your route. Each backend needs a matching expression that the gateway uses to match based on the request context in the selector at runtime.</p>
Specifies the type of the backend service	<p>Expand and specify policies as required for your environment.</p> <ul style="list-style-type: none"> Show route request policies Show route response policies Show response caching policies Show route logging policies

12. Review your selections on the Deployment page, then click **Create**.
Route 1 is configured.

Note that **HTTP** is selected and the **URL** field shows the REST Adapter-trigger based integration to later publish from your project in Oracle Integration.

The screenshot shows the Oracle Integration interface for configuring a route. The left sidebar has tabs for 'Basic information', 'Authentication', 'Routes' (which is selected), and 'Review'. The main area is titled 'Route 1' and contains the following fields:

- Path:** /order
- Methods:** POST
- Backend Type:** HTTP (selected)
- URL:** https://.../order
- Connection establishment timeout in seconds (Optional):** 60
- Request transmit timeout in seconds (Optional):** 10
- Reading response timeout in seconds (Optional):** 10
- Disable SSL verification:** (checkbox)

Below the main configuration, there are two buttons: 'Edit added single backend' and 'Edit added multiple backends'.

13. Create additional routes, as needed.

For this example, a second route is created with a second REST Adapter-trigger based integration to later publish from your project in Oracle Integration.

Route 2

Path: /query

Methods: GET

Backend Type: HTTP

URL: https://

Connection establishment timeout in seconds: 60

Request transmit timeout in seconds: 10

Reading response timeout in seconds: 10

Disable SSL verification

14. Follow Step 8 through Step 12 to create any additional deployments and routes. For this example, a second deployment (**Oracle Rest Say Hello World**) is created that includes a single route:

Route 1

Path: /names/{name}

Methods: GET

Backend Type: HTTP

URL: https://

Connection establishment timeout in seconds: 60

Request transmit timeout in seconds: 10

Reading response timeout in seconds: 10

Disable SSL verification

When complete, the two deployments are listed on the details page for the gateway.

The screenshot shows the Oracle Cloud Infrastructure API Gateway console. At the top, there's a header with 'Edit', 'Move resource', 'Add tags', and a red 'Delete' button. Below that is a 'Gateway information' section with fields like OCID, Compartment, Type (Public), Subnet, Hostname, IP address, Created, and Updated. Underneath is a 'Deployments' section with a 'Create deployment' button. A table lists two deployments: 'Netsuite' and 'Oracle Rest Say Hello World'. Each deployment has a status (Active), endpoint (e.g., ...m/netsuite), and deployment date (e.g., Wed, Nov 15, 2023, 05:49:49 UTC). To the right of the table is a small icon representing the deployment.

You are now ready to deploy individual integration endpoints as routes to Oracle Cloud Infrastructure API Gateway.

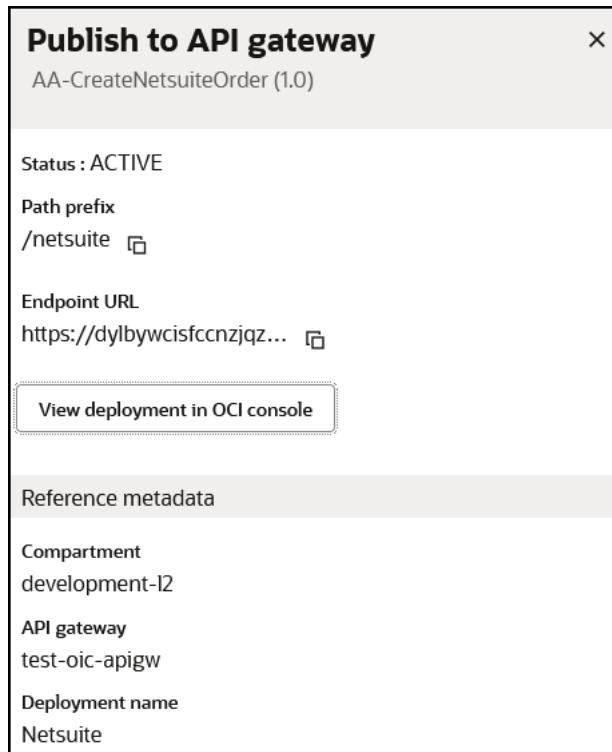
Deploy an Integration to API Gateway

After completing all prerequisites and gateway and deployment configuration tasks, you can deploy individual integration endpoints as routes from a project to Oracle Cloud Infrastructure API Gateway.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, find the already-activated integration to publish. Only active REST Adapter, triggered-based integrations can be deployed.
4. Click **Actions ...**, and select **Publish to API Gateway**.
The Publish to API gateway panel opens.
5. Enter the following information:

Element	Description
REST endpoint	Displays the endpoint of the integration. This endpoint cannot be deselected.
Search or select compartment	Select the compartment created in the Oracle Cloud Infrastructure Console that includes Oracle Cloud Infrastructure API Gateway.
Search or select API gateway	Select the Oracle Cloud Infrastructure API Gateway instance created in that compartment.
Select or create API gateway deployment	Select a deployment inside the gateway instance. You can also create a gateway from this field.

For this example, the **test-oic-apigw** gateway and **NetSuite** deployment created in [Create an Oracle Cloud Infrastructure API Gateway and Deployment in Oracle Cloud Infrastructure Console](#) are selected.



6. Click **Publish**.
The deployment is visible in Oracle Cloud Infrastructure API Gateway. Use the **Observe** tab in the project to check if the API calls are being reached when invoked by the Oracle Cloud Infrastructure API Gateway deployment URL.
7. If you want to unpublish your API gateway deployment, click **Undo publish**.

Clone an Integration in a Project

Cloning an integration in a project creates a new copy with identical connections and data mappings. The cloned version must have a unique identifier and/or version. You can also specify a name, keyword, and description. You cannot specify a package name. The remaining configuration is the same. You can reconfigure the clone after you create it. You can also clone a locked integration.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, find the integration to clone.
4. Click **Actions ...**, then select **Clone**.
The Clone integration panel opens.
5. Enter a name, unique identifier, version number, keywords, 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**.

Update the Tracing Level of Integrations in a Project

You can change the tracing level of an active integration without having to deactivate and reactivate it. You can also set the tracing level globally for all integrations in a project.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, hover over the active integration on which to set tracing.
4. Click **Actions . . .** and select **Configure activation**.
5. Follow the steps to set the tracing level. See [Manage Tracing Levels on Integrations](#).

Test REST Adapter Trigger Connection-Based Integrations in Projects

You can test application integrations in projects that are designed with a REST Adapter trigger connection.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, hover over the specific integration to run.
4. Click **Actions . . .** and select **Run**. The **Run** option is only available for REST Adapter trigger connection-based integrations.
The Configure and run page is displayed.
5. Follow the steps to run the integration. See [Test Integrations from Outside the Integration Canvas](#).

View the Dependent Relationships Between Project Resources

You can view the dependent relationships between all integrations, connections, lookups, JavaScript libraries, events, HL7 messages, and healthcare schemas in a project, in another project, or outside a project.

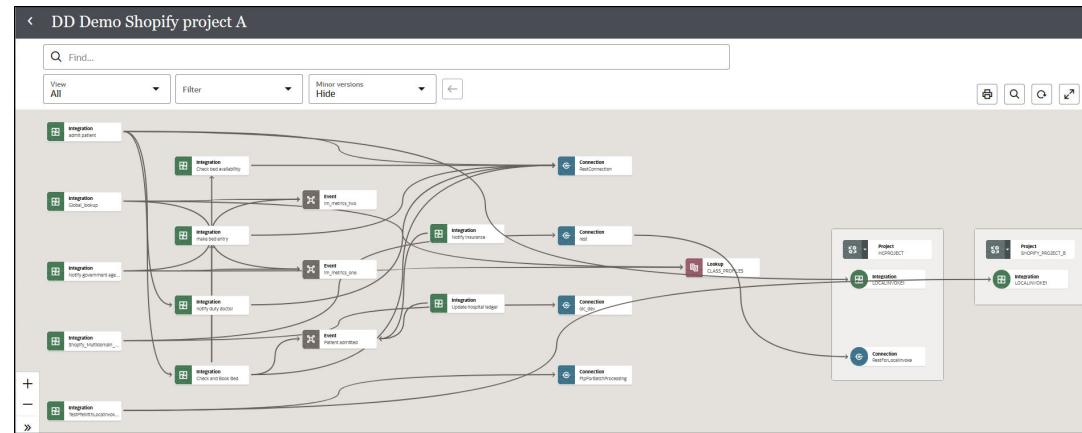
- [View Project Dependencies](#)
- [View a Project Summary Report](#)
- [View Lookup Usage](#)
- [View the Dependency Diagrams of Different Integration Versions](#)
- [View Integrations that are Shared Between Projects](#)

View Project Dependencies

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. Click **View dependencies** .

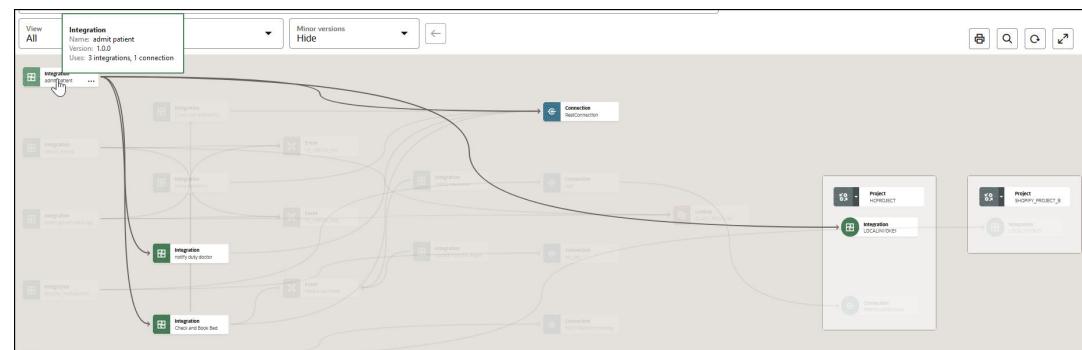
A view of the dependencies between all resources in the current project and other projects is displayed. If you are invoking an integration in a different project, that integration is shown in a container with the label **Project** and the project name as the container title. If

the diagram is dependent upon an integration that is outside of any project, the container is labeled **Not in project**. An option for printing the dependency diagram is available in the upper right corner.

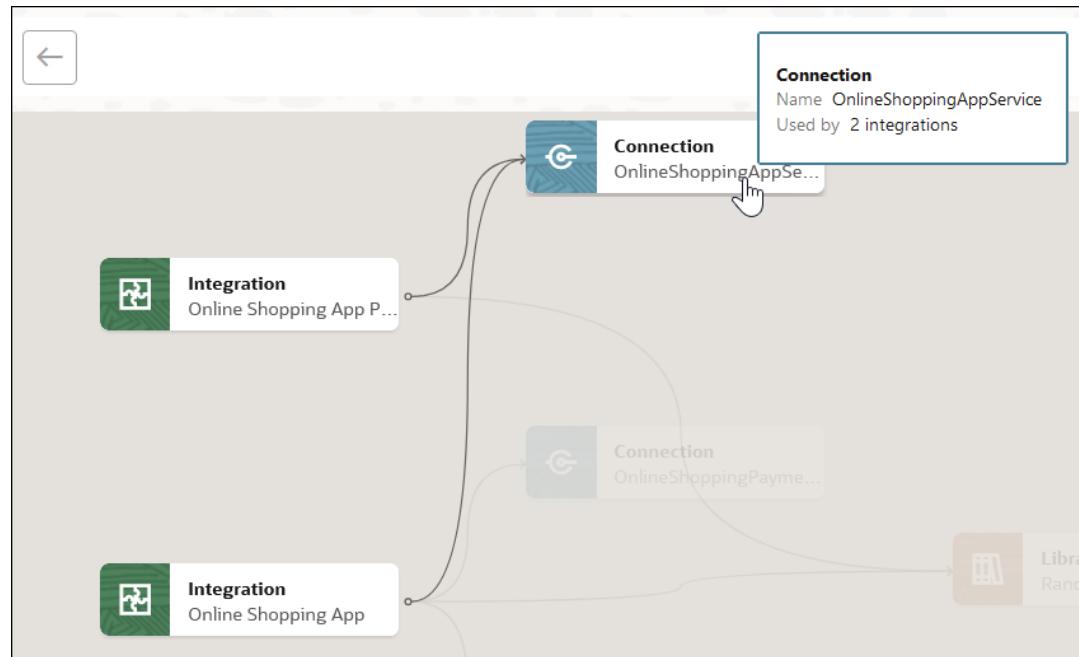


4. Hover over a resource to display its dependencies.

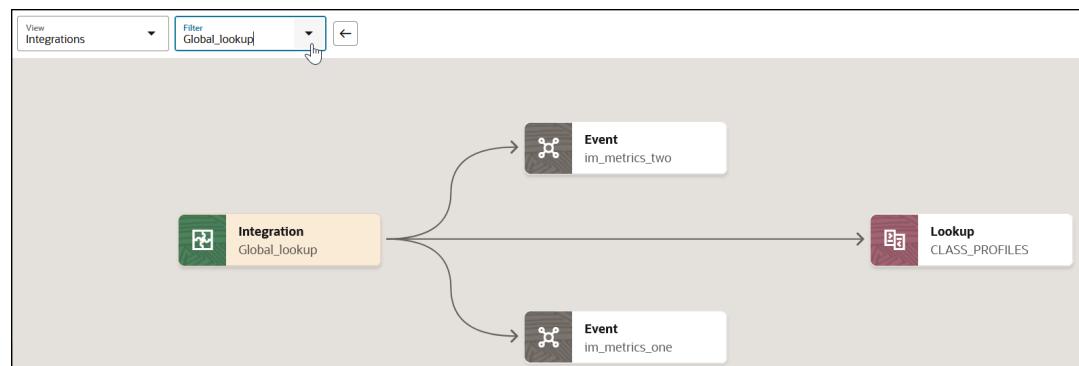
For this example, the integration uses one connection and three integrations, including one in another project. Hovering over the lines connecting the nodes also shows the relationship between the nodes in the tool tip.



5. Hover over a connection to see its dependent resources. For this example, the connection is used in two integrations.



6. Click the nodes to drill down into the resource-specific view, and keep clicking to drill down further.
7. Click **Back** ← to return to the previous view until you return to the beginning of the navigation.
8. From the **View** list, select to view specific resources (integrations, connections, lookups, JavaScript libraries, HL7 messages, healthcare schemas, and events). By default, **All** is selected to show all resources in the project.
9. From the **Filter** list, select to view a specific integration or connection and its dependencies. When you select a specific integration or connection, the **View** list changes to show the same resource type. For example:



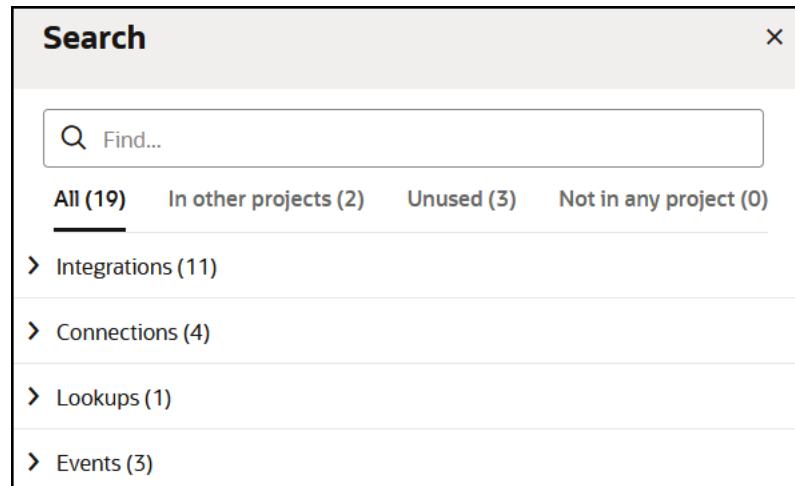
View a Project Summary Report

1. Click **View dependencies** .

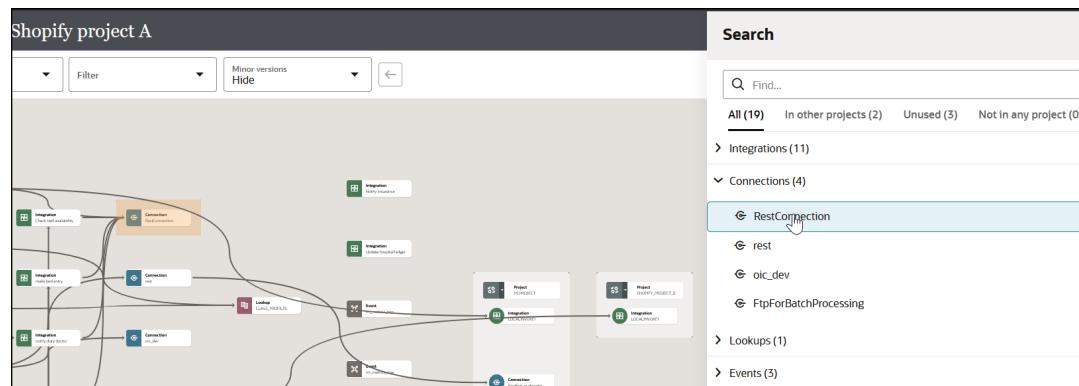
2. Click **Project summary** .

The Summary panel shows tabs for the dependent resources in the current project, the dependent resources being used in other projects, the unused dependent resources, and

the dependent resources being used that are not in projects. You can search on the name of a resource to display it.



3. Click a tab, and expand the resource category.
4. Click a specific resource to highlight its location in the dependency diagram.



View Lookup Usage

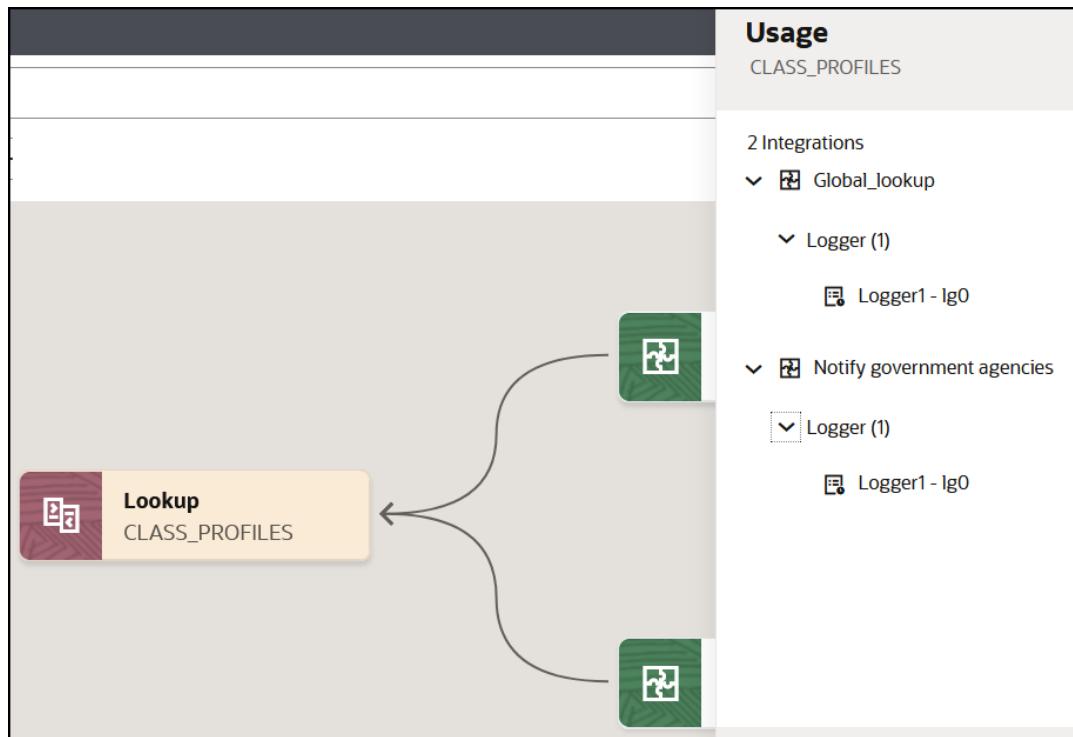
You can view the specific action in an integration in which a lookup is being used.

1. Click **View dependencies** .
2. Hover over a lookup in the dependency diagram to see the integrations in which it is used.



3. Click the lookup for a closer view.
4. In the lookup, click **Actions ...**, and select **Usage**.
The Usage panel is displayed.

5. Expand the integrations in the panel to see the actions using the lookup. For this example, log actions in two integrations are using the lookup.

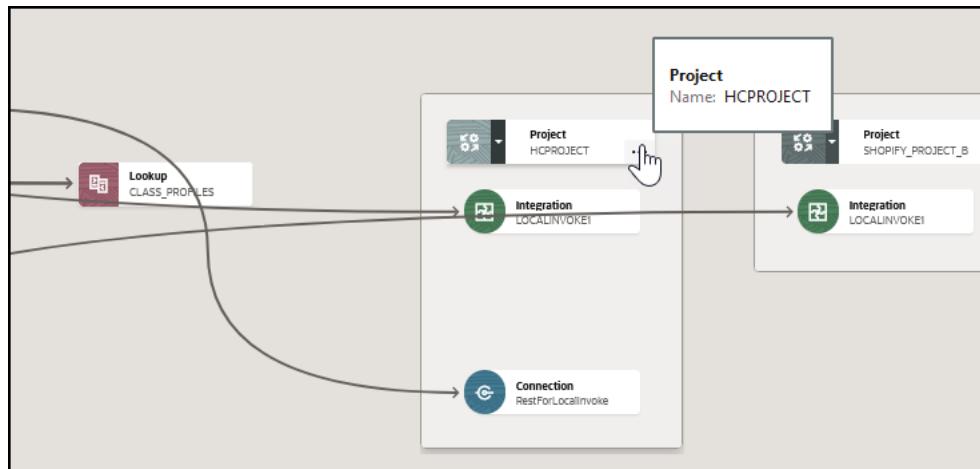


6. Hover over a log action name to display a copy icon for copying the name.
7. Go to the specific integration and click **Search**  to open the Search panel.
8. Enter the name to find its location in the integration.

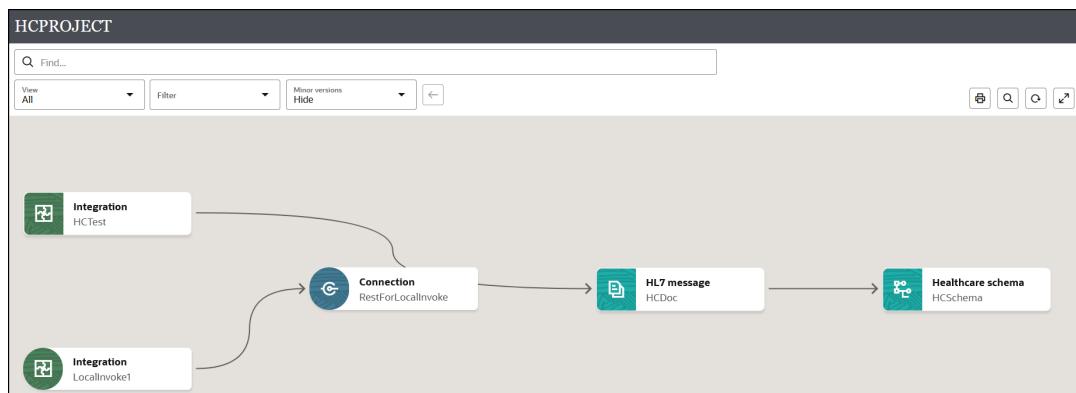
Navigate to the Dependency Diagrams of Remote Projects

If your dependency diagram includes dependent resources in other (remote) projects, you can access the dependency diagrams for those projects.

1. Click **View dependencies** .
2. Hover over a remote project in the dependency diagram. These projects are identified by the word **Project** and the project name.
3. Click **Actions ...**, and select **Open**.



The remote project opens in a second browser tab.



4. View resources in the second project, then close the browser tab when done.

View the Dependency Diagrams of Different Integration Versions

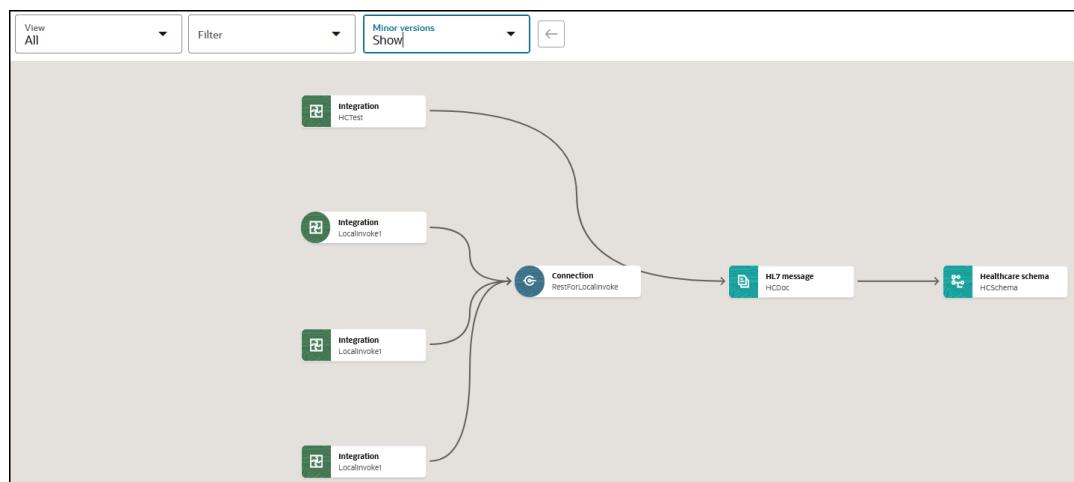
The latest active minor version of an integration is shown by default in the dependency diagram. You can also view all minor versions of an integration in the dependency diagram.

1. Click **View dependencies** for a project with multiple integration versions. For this example, **LocalInvoke1** has three minor versions.

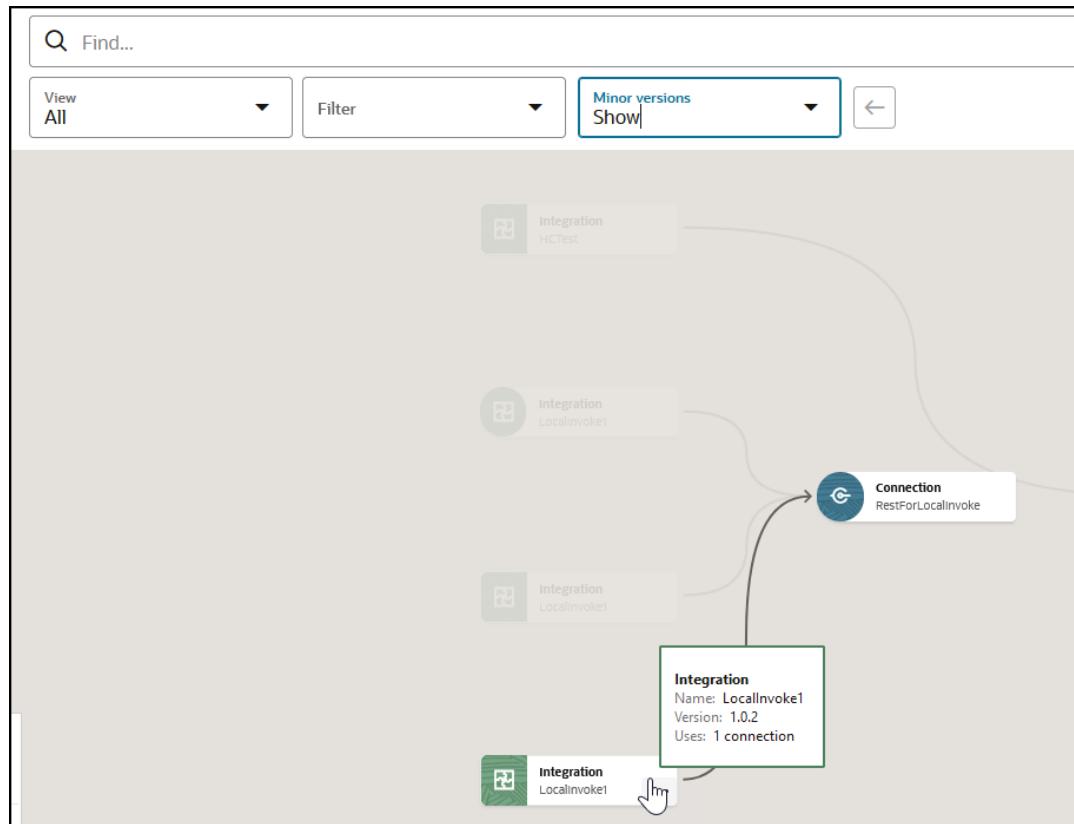
The screenshot shows the 'Integration' page with the following details:

- Deploy** and **Observe** buttons at the top left.
- A 'View dependencies' button with a magnifying glass icon at the top right.
- Integrations** section:
 - LocalInvoke1 (version 1.0.2, Configured)
 - LocalInvoke1 (version 1.0.1, Configured)
 - LocalInvoke1 (version 1.0.0, Active)
 - HCTest (version 1.0.0, Configured)
- Connections** section:
 - rest RestForLocalInvoke

- From the **Minor versions** list, select **Show**.



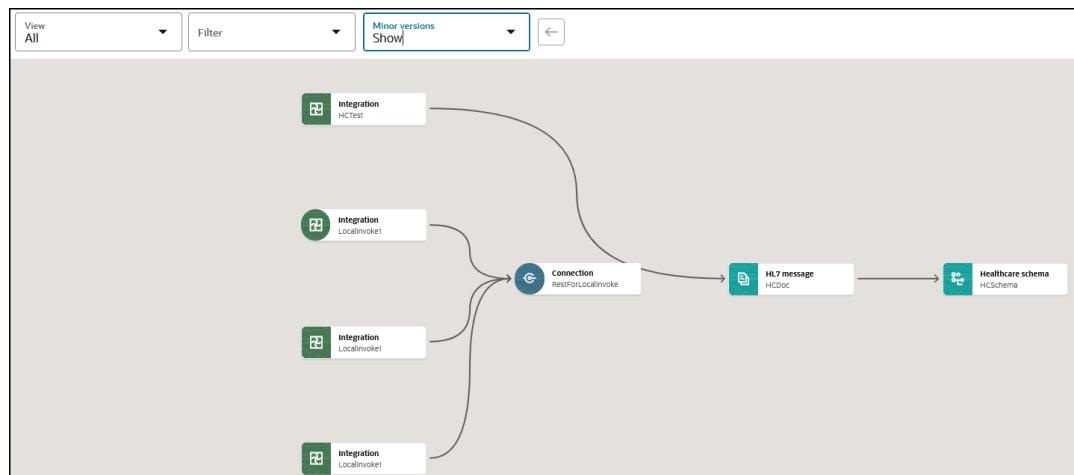
- Hover over the integrations to see the versions and their dependent resources. For this example, minor version 1.0.2 is shown.



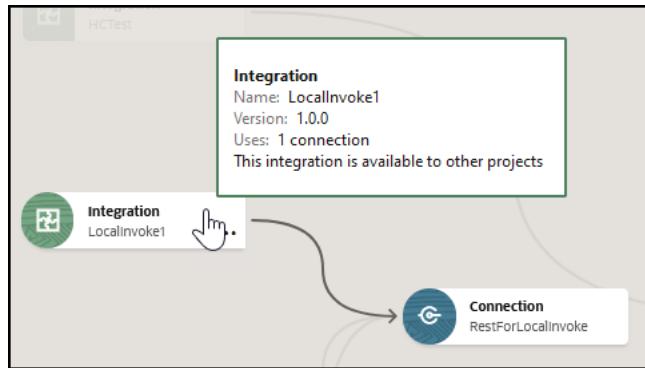
View Integrations that are Shared Between Projects

Integrations that are shared between projects are identified by a green circle in the dependency diagram.

1. Click **View dependencies** .
2. View the round circle for the second integration in the dependency diagram. This circle indicates that this integration is shared between projects. Integrations that are not shared with other projects are identified by a green square.



3. Hover over the shared integration and note that the tool tip indicates this integration is available to other projects.



View Run Details About Integrations in Projects

You can view run details about active application integrations in projects.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the **Integrations** section, hover over the specific integration to view.
4. Click **Actions ...** and select **Run details**. The metadata URL of the integration is displayed.

View Project Status

The overall status of a project is based on the status of its integrations.

1. In the navigation pane, click **Projects**.

The **Type** column indicates the project type is **Developed**, meaning it was created by a user.

2. View the overall status of the project in the **Status** column.

Overall project status is based on the status of each integration in your project. The following table provides example of how the overall status is determined. For this example, assume you have three integrations in your project.

If Integration 1 Status Is...	If Integration 2 Status Is...	If Integration 3 Status Is...	Then Project Status Is...
Draft	Draft	Draft	Draft
Draft	Draft	Configured	Draft
Draft	Configured	Configured	Draft
Configured	Configured	Configured	Configured
Configured	Configured	Active	Configured
Configured	Active	Active	Configured
Active	Active	Active	Active
Draft	Configured	Active	Draft

Edit Project Description Details

You can edit naming description details in user-developed projects that you specified during creation, such as the name, description, and keyword selections. You *cannot* change the project identifier value.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. In the upper right corner, click **Edit details** .
4. Optionally change the project name, description, and selected keywords.
5. Click **Save changes**.

Delete a Project

You can delete a project. However, you cannot delete a project if an integration is active or any resource is locked.

1. In the navigation pane, click **Projects**.
2. Hover over the project to delete.
3. Click **Actions ...**, then select **Delete**.

Monitor Integrations in a Project

You can track instances and monitor errors in a project from a single workspace during runtime. You can also access and view the activity stream that shows details about the movement of messages through integrations.

Topics:

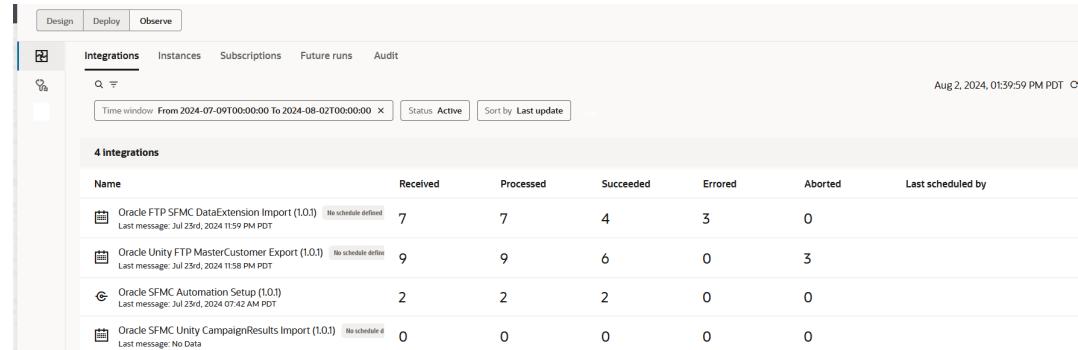
- [Monitor the Message Processing Status of Integrations in Projects](#)
- [Track Instances in a Project](#)
- [View Subscribed Instances in a Project](#)
- [Manage Errors in a Project](#)
- [View the Schedule Integration Runs Calendar in a Project](#)
- [View the Design-Time Audit Log of a Project](#)
- [Monitor Robots in a Project](#)

Monitor the Message Processing Status of Integrations in Projects

You can view the message processing status of your running integrations in a project, including how many messages have been received and processed, how many successful messages and errors have occurred, and how many messages have been canceled. You can also view aggregated instance metrics for the selected integration.

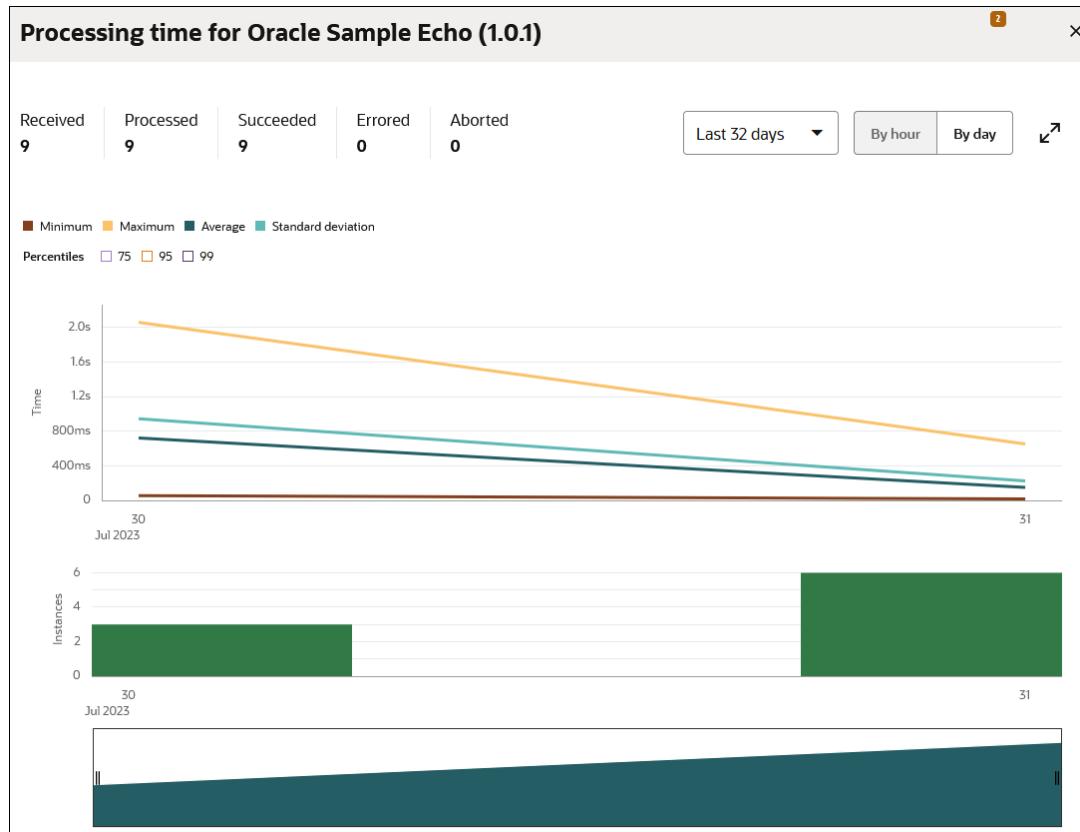
1. In the navigation pane, click **Projects**.

2. Click the project name or click **Edit** .
3. Click **Observe**.
4. Click **Integrations**.



Name	Received	Processed	Succeeded	Errorred	Aborted	Last scheduled by
Oracle FTP SFMC DataExtension Import (1.0.1)	7	7	4	3	0	
Oracle Unity FTP MasterCustomer Export (1.0.1)	9	9	6	0	3	
Oracle SFMC Automation Setup (1.0.1)	2	2	2	0	0	
Oracle SFMC Unity CampaignResults Import (1.0.1)	0	0	0	0	0	

5. Click **Filter** , then select the time window, integration status (active, configured, activating, failed, or all), and integration style (application, schedule, or event). You can also specify a custom time range to filter the search for integrations under **Date range** with the **From** and **To** buttons.
6. Click numbers in the **Received**, **Processed**, **Succeeded**, **Errorred**, and **Aborted** columns to access the list of instances in those states on the Instances tab of the project. If you click the **Received** number, instances are displayed based on their creation date. All other clickable numbers are sorted by their last updated date.
7. Hover over a row for an integration and click **View statistics**  to show aggregated instance metrics for the selected integration. For information about using the Statistics page, see [Step 6 in Monitor the Message Processing Status of Integrations](#).



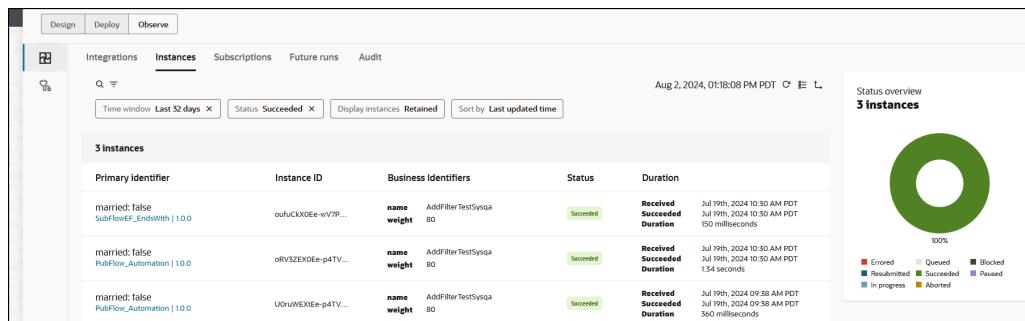
Track Instances in a Project

You can filter and track the status of integration instances through business identifiers. You can also view the activity stream and message payload of an instance you are tracking and replay integration instances configured as replayable.

1. In the navigation pane, click **Projects**.
2. Click the project name or click **Edit** .
3. Click **Observe**.
4. Click **Instances**.

The status of each integration instance is displayed in a graph. Each potential integration status is identified by a specific color.

5. View a specific status of your integration instances (for this example, **Succeeded**) in either of the following ways:
 - a. View the graph to identify if there are successfully completed instances and click that part.
 - b. Select **Succeeded** from the **Status** field in the **Filter**  icon.



6. Hover over the row of an integration instance and click **View Details** to access the activity stream.
7. Click **Download** to download the logs. The Oracle Cloud identifier (OCID) value of your service instance is also included in the logs.
8. Hover over the row of an integration instance and click **Actions** ..., and then **Run again** to replay REST Adapter-triggered and AS2 Adapter-triggered integration instances configured as replayable. This feature is available both inside and outside of projects. See [Capabilities and Restrictions](#) and [Understand the Differences Between Replaying and Recovering Instances](#).
9. Hover over the row of an integration instance and click **Actions** ..., and then **View Child Instances** to display child instances invoked by parent instances.
10. To the right of the time stamp, click **Chart View** .
- a. View time lines for instances grouped by integration. The scale is shown in minutes by default.

The following icons are displayed:

 - **Bar** icon: Indicates an instance has completed.
 - **Diamond** icon: Indicates an instance is incomplete (for example, has not started or is still running).
- b. Hover over either icon to display details such as instance ID and time stamp.
- c. Click **Seconds** to show the scale in seconds.
- d. Click **Expand** to show where there is overlap between completed instances and incomplete instances.
- e. Click **Collapse**.
11. Click **Back** to return to the **Observe** tab.

See [Track Integration Instances](#).

View Subscribed Instances in a Project

You can monitor the number of integrations that subscribe to events, and their status, on the Subscriptions page in a project.

1. In the navigation pane, click **Projects**.
2. Click the project name or click **Edit** .

3. Click **Observe**.
4. Click **Subscriptions**.

The Subscriptions page shows the active event subscriptions in the project.

5. Click **Filter**  to filter by event subscription integrations or retained events (events published, but not yet received by the event subscription integrations).

Manage Errors in a Project

You can manage errors that occur in integrations in a project.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. Click **Observe**.
4. Click **Instances**.
5. Identify if your project has errors in either of the following ways:
 - a. View the graph to identify if there are errors and click that part. Each potential integration status is identified by a specific color.
 - b. Select **Errored** from the **Status** field in the **Filter**  icon.

The page is refreshed to display tracking information about errors.

6. Hover over the row of an integration instance in error and click **View Details**  to access the activity stream.
- You can view error messages and the message payload in the activity stream of failed integration instances. You can also download the activity stream logs and the payload of the failed section of an integration instance. See [View Error Messages and the Message Payload in the Activity Stream of Failed Instances](#).
7. Hover over the row of an integration instance in error and click **Actions ...** to display other options:

If You Select This Option...	Then ...
Resubmit	Respond when prompted to resubmit the message. See Resubmit Failed Messages .
Abort	Respond when prompted to abort the message. See Discard Errors .
Schedule	The Schedule and future runs page is displayed. See View Future Schedule Integration Runs .
View Child Instances	View the child instances invoked by parent instances.

8. Click **Back**  to return to the **Observe** tab.

See [Manage Errors](#).

View the Schedule Integration Runs Calendar in a Project

You can view a calendar of all runs scheduled to trigger in the project in the future, across all schedule integrations whose schedules have started.

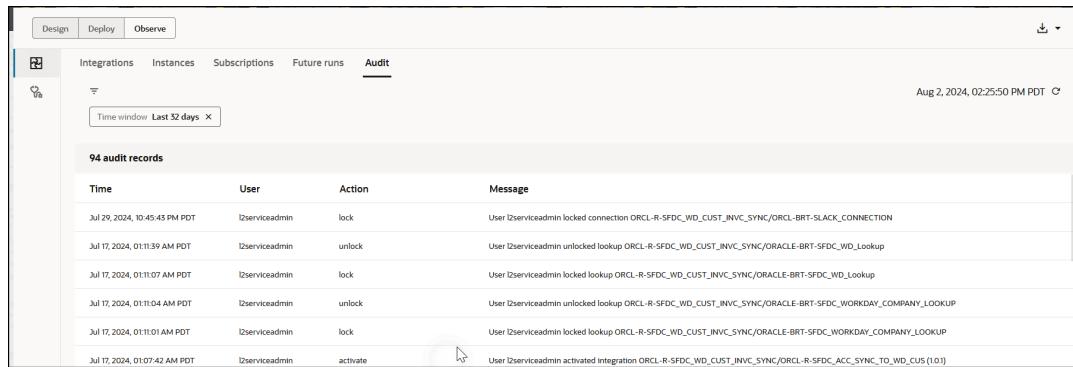
1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. Click **Observe**.
4. Click **Future runs**. See [View the Calendar of Schedule Integration Runs](#).

View the Design-Time Audit Log of a Project

You can view and download the design-time audit log of actions performed in a project.

1. In the navigation pane, click **Projects**.
2. Click the project name or click .
3. Click **Observe**.
4. Click **Audit**.

Each row represents a record. The ZIP file can contain a maximum of 10,000 audit records.



Time	User	Action	Message
Jul 29, 2024, 10:45:45 PM PDT	l2serviceadmin	lock	User l2serviceadmin locked connection ORCL-R-SFDC_WD_CUST_INVC_SYNC/ORACLE-BRT-SLACK_CONNECTION
Jul 17, 2024, 01:11:39 AM PDT	l2serviceadmin	unlock	User l2serviceadmin unlocked lookup ORCL-R-SFDC_WD_CUST_INVC_SYNC/ORACLE-BRT-SFDC_WD_Lookup
Jul 17, 2024, 01:11:07 AM PDT	l2serviceadmin	lock	User l2serviceadmin locked lookup ORCL-R-SFDC_WD_CUST_INVC_SYNC/ORACLE-BRT-SFDC_WD_Lookup
Jul 17, 2024, 01:11:04 AM PDT	l2serviceadmin	unlock	User l2serviceadmin unlocked lookup ORCL-R-SFDC_WD_CUST_INVC_SYNC/ORACLE-BRT-SFDC_WORKDAY_COMPANY_LOOKUP
Jul 17, 2024, 01:11:01 AM PDT	l2serviceadmin	lock	User l2serviceadmin locked lookup ORCL-R-SFDC_WD_CUST_INVC_SYNC/ORACLE-BRT-SFDC_WORKDAY_COMPANY_LOOKUP
Jul 17, 2024, 01:07:42 AM PDT	l2serviceadmin	activate	User l2serviceadmin activated integration ORCL-R-SFDC_WD_CUST_INVC_SYNC/ORCL-R-SFDC_ACC_SYNC_TO_WD_CUS (1.0)

5. Click **Filter**  to filter data by time window, integration name, user name, or actions such as activate and deactivate. You can click the **Data range** option to specify a date range that is greater than 32 days. The maximum date range is six months. If you are using projects, the project identifier is also included in the audit logs. Use this identifier to audit instance and error details.
6. Click **Download all** to download a ZIP file of the entire audit log.

For example:

```
[2023-03-23 06:20:11.554 Z] [userID: myserviceadmin]
[myserviceadmin,CREATE,ICS_AppInstance,OCI_NATIVE_FUNCTIONS_CONNECTION,OCI_
NATIVE_FUNCTIONS_CONNECTION,null] User myserviceadmin created Connection
OCI_NATIVE_FUNCTIONS_CONNECTION
[2023-03-10 18:05:48.559 Z] [userID: myserviceadmin]
```

```
[myserviceadmin,CREATE,ICS_AppInstance,APPLICATION_EVENT_CONNECTION,APPLICATION_EVENT_CONNECTION,null] User myserviceadmin created Connection APPLICATION_EVENT_CONNECTION  
[2023-03-10 18:05:48.349 Z] [userID: myserviceadmin]  
[myserviceadmin,CREATE,ICS_AppInstance,STAGE_FILE_CONNECTION,STAGE_FILE_CONNECTION,null] User myserviceadmin created Connection STAGE_FILE_CONNECTION  
[2023-03-10 18:05:48.158 Z] [userID: myserviceadmin]  
[myserviceadmin,CREATE,ICS_AppInstance,PRESEEDED_B2B_TRANSLATOR_CONN_1851,B2B,null] User myserviceadmin created Connection B2B  
[2023-03-10 18:05:47.750 Z] [userID: myserviceadmin]  
[myserviceadmin,CREATE,ICS_AppInstance,PRESEEDED_COLLOCATED_CONN_1741,Local Integration,null] User myserviceadmin created Connection Local Integration
```

7. Click **Download filtered** to download records based on the currently-selected filters that you configured with the **Filter**  option (data range, integration name, user name, and action).

Monitor Robots in a Project

You can monitor robot instances directly within a project.

All information about robots is in a separate guide. See Monitor an Automation Solution in *Using Robots in Oracle Integration 3*.

4

Create Connections

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

Topics:

- [Inbound and Outbound Endpoints and Transport Layer Security Server Version Support](#)
- [Create a Connection](#)
- [Configure Connection Properties and Security Properties](#)
- [Test the Connection](#)
- [Edit a Connection](#)
- [Clone a Connection](#)
- [Delete a Connection](#)
- [Unlock a Connection](#)
- [Refresh Integration Metadata](#)
- [Manage Security Certificates](#)
- [View Preinstalled Adapters](#)
- [Allow JSON Numbers with High Precision and Scale](#)

Inbound and Outbound Endpoints and Transport Layer Security Server Version Support

All inbound endpoints for integrations in Oracle Integration are hosted on SSL servers that can accept requests coming from clients supporting transport layer security (TLS) 1.2. This is true regardless of whether they are SOAP- or REST-enabled and regardless of the adapter used as the trigger connection. TLS version 1.3 is supported for outbound integrations.

Create a Connection

Before you can build an integration, you must create the connections to the applications with which you want to share data.

To create a connection in Oracle Integration:

1. In the navigation pane, click **Design**, then **Connections**.
2. Click **Create**.

 Note:

You can also create a connection in the integration canvas. See Define Inbound Triggers and Outbound Invokes.

3. In the Create connection panel, select the adapter to use for this connection. To find the adapter, scroll through the list, or enter a partial or full name in the **Search** field.
4. Enter the information that describes this connection.

Element	Description
Name	Enter a meaningful name to help others find your connection when they begin to create their own integrations.
Identifier	Automatically displays the name in capital letters that you entered in the Name field. If you modify the identifier name, don't include blank spaces (for example, SALES OPPORTUNITY).
Role	Select the role (direction) in which to use this connection (trigger, invoke, or both). Only the roles supported by the adapter are displayed for selection. When you select a role, only the connection properties and security policies appropriate to that role are displayed on the Connections page. If you select an adapter that supports both invoke and trigger, but select only one of those roles, you'll get an error when you try to drag the adapter into the section you didn't select. For example, assume you configure a connection for the Oracle Service Cloud (RightNow) Adapter as only an invoke . Dragging the adapter to a trigger section in the integration produces an error.
Keywords	Enter optional keywords (tags). You can search on the connection keywords on the Connections page.
Description	Enter an optional description of the connection.
Share with other projects	Note: This field only appears if you are creating a connection in a project. Select to make this connection publicly available in other projects. Connection sharing eliminates the need to create and maintain separate connections in different projects. When you configure an adapter connection in a different project, the Use a shared connection field is displayed at the top of the Connections page. If the connection you are configuring matches the same type and role as the publicly available connection, you can select that connection to reference (inherit) its resources. See Add and Share a Connection Across a Project.

5. Click **Create**.

Your connection is created. You're now ready to configure the connection properties, security policies, and (for some connections) access type.

Configure Connection Properties and Security Properties

Enter connection and security property information for the adapter.

Configure Connection Properties

Enter connection information so your application can process requests.

1. Go to the **Properties** section.
2. See the following sections for information about specifying connection properties for the adapter you selected.

Adapter	For Information
Adobe Sign Adapter	<i>Using the Adobe Sign Adapter with Oracle Integration 3</i>
Apache Kafka Adapter	<i>Using the Apache Kafka Adapter with Oracle Integration 3</i>
<i>Using the ArcGIS (ESRI) Adapter with Oracle Integration 3</i>	<i>Using the ArcGIS (ESRI) Adapter with Oracle Integration 3</i>
AS2 Adapter	<i>Using the AS2 Adapter with Oracle Integration 3</i>
AS4 Adapter	<i>Using the AS4 Adapter with Oracle Integration 3</i>
Amazon Simple Notification Service (SNS) Adapter	<i>Using the Amazon Simple Notification Service (SNS) Adapter with Oracle Integration 3</i>
Amazon Simple Queue Service (SQS) Adapter	<i>Using the Amazon Simple Queue Service (SQS) Adapter with Oracle Integration 3</i>
Amazon Simple Storage Service (S3) Adapter	<i>Using the Amazon Simple Storage Service (S3) Adapter with Oracle Integration 3</i>
Azure Active Directory Adapter	<i>Using the Azure Active Directory Adapter with Oracle Integration 3</i>
Azure Event Grid Adapter	<i>Using the Azure Event Grid Adapter with Oracle Integration 3</i>
Azure Service Bus Adapter	<i>Using the Azure Service Bus Adapter with Oracle Integration 3</i>
Azure Storage Adapter	<i>Using the Azure Storage Adapter with Oracle Integration 3</i>
Box Adapter	N/A
Confluent Adapter	<i>Using the Confluent Adapter with Oracle Integration 3</i>
Coupa Procurement Adapter	<i>Using the Coupa Procurement Adapter with Oracle Integration 3</i>
Coupa Sourcing Adapter	<i>Using the Coupa Sourcing Adapter with Oracle Integration 3</i>
DocuSign Adapter	<i>Using the DocuSign Adapter with Oracle Integration 3</i>
eBay Marketplace Adapter	<i>Using the eBay Marketplace Adapter with Oracle Integration 3</i>
Eventbrite Adapter	<i>Using the Eventbrite Adapter with Oracle Integration 3</i>

Adapter	For Information
Facebook Adapter	<i>Using the Facebook Adapter with Oracle Integration 3</i>
FHIR Adapter	<i>Using the FHIR Adapter with Oracle Integration 3</i>
File Adapter	<i>Using the File Adapter with Oracle Integration 3</i>
FTP Adapter	<i>Using the FTP Adapter with Oracle Integration 3</i>
GCP Pub Sub Adapter	<i>Using the GCP Pub Sub Adapter with Oracle Integration 3</i>
GCP Storage Adapter	<i>Using the GCP Storage Adapter with Oracle Integration 3</i>
GitHub Adapter	N/A
GraphQL Adapter	<i>Using the GraphQL Adapter with Oracle Integration 3</i>
Google Gmail Adapter	<i>Using the Google Gmail Adapter with Oracle Integration 3</i>
Google Calendar Adapter	<i>Using the Google Calendar Adapter with Oracle Integration 3</i>
Google Tasks Adapter	<i>Using the Google Tasks Adapter with Oracle Integration 3</i>
HubSpot Adapter	<i>Using the HubSpot Adapter with Oracle Integration 3</i>
IBM DB2 Adapter	<i>Using the IBM DB2 Adapter with Oracle Integration 3</i>
IBM MQ Series JMS Adapter	<i>Using the IBM MQ Series JMS Adapter with Oracle Integration 3</i>
Jira Adapter	<i>Using the Jira Adapter with Oracle Integration 3</i>
LinkedIn Adapter	<i>Using the LinkedIn Adapter with Oracle Integration 3</i>
Magento Commerce Cloud Adapter	<i>Using the Magento Commerce Cloud Adapter with Oracle Integration 3</i>
Mailchimp Adapter	<i>Using the Mailchimp Adapter with Oracle Integration 3</i>
Marketo Adapter	<i>Using the Marketo Adapter with Oracle Integration 3</i>
Microsoft Office 365 Calendar Adapter	<i>Using the Microsoft Office 365 Calendar Adapter with Oracle Integration 3</i>
Microsoft Office 365 People Adapter	<i>Using the Microsoft Office 365 People Adapter with Oracle Integration 3</i>
Microsoft Office 365 Outlook Adapter	<i>Using the Microsoft Office 365 Outlook Adapter with Oracle Integration 3</i>
Microsoft SQL Server Adapter	<i>Using the Microsoft SQL Server Adapter with Oracle Integration 3</i>
MLLP Adapter	<i>Using the MLLP Adapter with Oracle Integration 3</i>
MS Dynamics 365 CRM Adapter	<i>Using the MS Dynamics 365 CRM Adapter with Oracle Integration 3</i>
MySQL Adapter	<i>Using the MySQL Adapter with Oracle Integration 3</i>
Netezza Adapter	<i>Using the Netezza Adapter with Oracle Integration 3</i>
OData Adapter	<i>Using the OData Adapter with Oracle Integration 3</i>
Oracle Aconex Cloud Adapter	<i>Using the Oracle Aconex Cloud Adapter with Oracle Integration 3</i>

Adapter	For Information
Oracle Advanced Queuing (AQ) Adapter	<i>Using the Oracle Advanced Queuing (AQ) Adapter with Oracle Integration 3</i>
Oracle Autonomous Data Warehouse Adapter	<i>Using the Oracle Autonomous Data Warehouse Adapter with Oracle Integration 3</i>
Oracle Autonomous Transaction Processing Adapter	<i>Using the Oracle Autonomous Transaction Processing Adapter with Oracle Integration 3</i>
Oracle Cloud Infrastructure Streaming Service Adapter	<i>Using the Oracle Cloud Infrastructure Streaming Service Adapter with Oracle Integration 3</i>
Oracle Commerce Cloud Adapter	<i>Using the Oracle Commerce Cloud Adapter with Oracle Integration 3</i>
Oracle CPQ Adapter	<i>Using the Oracle CPQ Adapter with Oracle Integration 3</i>
Oracle Database Adapter	<i>Using the Oracle Database Adapter with Oracle Integration 3</i>
Oracle Database Cloud Service Adapter	<i>Using the Oracle Database Cloud Service Adapter with Oracle Integration 3</i>
Oracle E-Business Suite Adapter	<i>Using the Oracle E-Business Suite Adapter with Oracle Integration 3</i>
Oracle Eloqua Cloud Adapter	N/A
Oracle CX Sales and B2B Service Adapter	<i>Using the Oracle CX Sales and B2B Service Adapter with Oracle Integration 3</i>
Oracle Enterprise Performance Management Cloud Adapter	<i>Using the Oracle Enterprise Performance Management Cloud Adapter with Oracle Integration 3</i>
Oracle ERP Cloud Adapter	<i>Using the Oracle ERP Cloud Adapter with Oracle Integration 3</i>
Oracle Field Service Cloud Adapter	<i>Using the Oracle Field Service Cloud Adapter with Oracle Integration 3</i>
Oracle HCM Cloud Adapter	<i>Using the Oracle HCM Cloud Adapter with Oracle Integration 3</i>
Oracle Hospitality Adapter	<i>Using the Oracle Hospitality Adapter with Oracle Integration 3</i>
Oracle Intelligent Advisor Adapter	<i>Using the Oracle Intelligent Advisor Adapter with Oracle Integration 3</i>
Oracle Intelligent Track and Trace Adapter	<i>Using the Oracle Intelligent Track and Trace Adapter with Oracle Integration 3</i>
Oracle JD Edwards EnterpriseOne Adapter	<i>Using the Oracle JD Edwards EnterpriseOne Adapter with Oracle Integration 3</i>
Oracle Logistics Adapter	<i>Using the Oracle Logistics Adapter with Oracle Integration 3</i>
Oracle NetSuite Adapter	<i>Using the Oracle NetSuite Adapter with Oracle Integration 3</i>
Oracle Primavera Cloud Adapter	<i>Using the Oracle Primavera Cloud Adapter with Oracle Integration 3</i>
Oracle Primavera Cloud Data Service Adapter	<i>Using the Oracle Primavera Cloud Data Service Adapter with Oracle Integration 3</i>
Oracle Primavera P6 EPPM Adapter	<i>Using the Oracle Primavera P6 EPPM Adapter with Oracle Integration 3</i>
Oracle Responsys Adapter	<i>Using the Oracle Responsys Adapter with Oracle Integration 3</i>
Oracle Service Cloud (RightNow) Adapter	<i>Using the Oracle Service Cloud (RightNow) Adapter with Oracle Integration 3</i>

Adapter	For Information
Oracle Siebel Adapter	<i>Using the Oracle Siebel Adapter with Oracle Integration 3</i>
Oracle SOA Suite Adapter	<i>Using the Oracle SOA Suite Adapter with Oracle Integration 3</i>
Oracle Taleo Enterprise Edition Adapter	<i>Using the Oracle Taleo Enterprise Edition Adapter with Oracle Integration 3</i>
Oracle Unity Adapter	<i>Using the Oracle Unity Adapter with Oracle Integration</i>
Oracle Utilities Adapter	<i>Using the Oracle Utilities Adapter with Oracle Integration 3</i>
Oracle WebLogic JMS Adapter	<i>Using the Oracle WebLogic JMS Adapter with Oracle Integration 3</i>
PayPal Adapter	<i>Using the PayPal Adapter with Oracle Integration 3</i>
PostgreSQL Adapter	<i>Using the PostgreSQL Adapter with Oracle Integration 3</i>
Primavera Unifier Adapter	<i>Using the Oracle Primavera Unifier Adapter with Oracle Integration 3</i>
REST Adapter	<i>Using the REST Adapter with Oracle Integration 3</i>
RosettaNet Adapter	<i>Using the RosettaNet Adapter with Oracle Integration 3</i>
Salesforce Adapter	<i>Using the Salesforce Adapter with Oracle Integration 3</i>
Salesforce Commerce Cloud Adapter	<i>Using the Salesforce Commerce Cloud Adapter with Oracle Integration 3</i>
Salesforce Marketing Cloud Adapter	<i>Using the Salesforce Marketing Cloud Adapter with Oracle Integration 3</i>
Salesforce REST Adapter	<i>Using the Salesforce REST Adapter with Oracle Integration 3</i>
SAP Adapter	<i>Using the SAP Adapter with Oracle Integration 3</i>
SAP Ariba Adapter	<i>Using the SAP Ariba Adapter with Oracle Integration 3</i>
SAP ASE (Sybase) Adapter	<i>Using the SAP ASE (Sybase) Adapter with Oracle Integration 3</i>
SAP Commerce Cloud (Hybris) Adapter	<i>Using the SAP Commerce Cloud (Hybris) Adapter with Oracle Integration 3</i>
SAP Concur Adapter	<i>Using the SAP Concur Adapter with Oracle Integration 3</i>
SAP SuccessFactors Adapter	<i>Using the SAP SuccessFactors Adapter with Oracle Integration 3</i>
SAP S/4HANA Cloud Adapter	<i>Using the SAP S/4HANA Cloud Adapter with Oracle Integration 3</i>
ServiceNow Adapter	<i>Using the ServiceNow Adapter with Oracle Integration 3</i>
Shopify Adapter	<i>Using the Shopify Adapter with Oracle Integration 3</i>
Shopify GraphQL Adapter	<i>Using the Shopify GraphQL Adapter with Oracle Integration 3</i>
Slack Adapter	N/A
Snowflake Adapter	<i>Using the Snowflake Adapter with Oracle Integration 3</i>

Adapter	For Information
SOAP Adapter	<i>Using the SOAP Adapter with Oracle Integration 3</i>
SugarCRM Adapter	<i>Using the SugarCRM Adapter with Oracle Integration 3</i>
SurveyMonkey Adapter	<i>Using the SurveyMonkey Adapter with Oracle Integration 3</i>
Trello Adapter	<i>Using the Trello Adapter with Oracle Integration 3</i>
Twilio Adapter	<i>Using the Twilio Adapter with Oracle Integration 3</i>
Twitter Adapter	<i>Using the Twitter Adapter with Oracle Integration 3</i>
Workday Adapter	<i>Using the Workday Adapter with Oracle Integration 3</i>
Zendesk Adapter	<i>Using the Zendesk Adapter with Oracle Integration 3</i>
Zuora Adapter	<i>Using the Zuora Adapter with Oracle Integration 3</i>

You are now ready to configure connection security.

Configure Connection Security

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

1. Go to the **Security** section.
2. See the following sections for information about selecting the security policy and entering your login credentials.

Adapter	For Information
Adobe Sign Adapter	<i>Using the Adobe Sign Adapter with Oracle Integration 3</i>
Apache Kafka Adapter	<i>Using the Apache Kafka Adapter with Oracle Integration 3</i>
Using the ArcGIS (ESRI) Adapter with Oracle Integration 3	<i>Using the ArcGIS (ESRI) Adapter with Oracle Integration 3</i>
AS2 Adapter	<i>Using the AS2 Adapter with Oracle Integration 3</i>
AS4 Adapter	<i>Using the AS4 Adapter with Oracle Integration 3</i>
Amazon Simple Notification Service (SNS) Adapter	<i>Using the Amazon Simple Notification Service (SNS) Adapter with Oracle Integration 3</i>
Amazon Simple Queue Service (SQS) Adapter	<i>Using the Amazon Simple Queue Service (SQS) Adapter with Oracle Integration 3</i>
Amazon Simple Storage Service (S3) Adapter	<i>Using the Amazon Simple Storage Service (S3) Adapter with Oracle Integration 3</i>
Azure Active Directory Adapter	<i>Using the Azure Active Directory Adapter with Oracle Integration 3</i>
Azure Event Grid Adapter	<i>Using the Azure Event Grid Adapter with Oracle Integration 3</i>
Azure Service Bus Adapter	<i>Using the Azure Service Bus Adapter with Oracle Integration 3</i>
Azure Storage Adapter	<i>Using the Azure Storage Adapter with Oracle Integration 3</i>
Box Adapter	<i>Using the Box Adapter with Oracle Integration 3</i>

Adapter	For Information
Confluent Adapter	<i>Using the Confluent Adapter with Oracle Integration 3</i>
Coupa Procurement Adapter	<i>Using the Coupa Procurement Adapter with Oracle Integration 3</i>
Coupa Sourcing Adapter	<i>Using the Coupa Sourcing Adapter with Oracle Integration 3</i>
DocuSign Adapter	<i>Using the DocuSign Adapter with Oracle Integration 3</i>
eBay Marketplace Adapter	<i>Using the eBay Marketplace Adapter with Oracle Integration 3</i>
Eventbrite Adapter	<i>Using the Eventbrite Adapter with Oracle Integration 3</i>
Facebook Adapter	<i>Using the Facebook Adapter with Oracle Integration 3</i>
FHIR Adapter	<i>Using the FHIR Adapter with Oracle Integration 3</i>
File Adapter	NA
FTP Adapter	<i>Using the FTP Adapter with Oracle Integration 3</i>
GCP Pub Sub Adapter	<i>Using the GCP Pub Sub Adapter with Oracle Integration 3</i>
GCP Storage Adapter	<i>Using the GCP Storage Adapter with Oracle Integration 3</i>
GitHub Adapter	<i>Using the GitHub Adapter with Oracle Integration 3</i>
GraphQL Adapter	<i>Using the GraphQL Adapter with Oracle Integration 3</i>
Google Gmail Adapter	<i>Using the Google Gmail Adapter with Oracle Integration 3</i>
Google Calendar Adapter	<i>Using the Google Calendar Adapter with Oracle Integration 3</i>
Google Tasks Adapter	<i>Using the Google Tasks Adapter with Oracle Integration 3</i>
HubSpot Adapter	<i>Using the HubSpot Adapter with Oracle Integration 3</i>
IBM DB2 Adapter	<i>Using the IBM DB2 Adapter with Oracle Integration 3</i>
IBM MQ Series JMS Adapter	<i>Using the IBM MQ Series JMS Adapter with Oracle Integration 3</i>
Jira Adapter	<i>Using the Jira Adapter with Oracle Integration 3</i>
LinkedIn Adapter	<i>Using the LinkedIn Adapter with Oracle Integration 3</i>
Magento Commerce Cloud Adapter	<i>Using the Magento Commerce Cloud Adapter with Oracle Integration 3</i>
Mailchimp Adapter	<i>Using the Mailchimp Adapter with Oracle Integration 3</i>
Marketo Adapter	<i>Using the Marketo Adapter with Oracle Integration 3</i>
Microsoft Office 365 Calendar Adapter	<i>Using the Microsoft Office 365 Calendar Adapter with Oracle Integration 3</i>
Microsoft Office 365 People Adapter	<i>Using the Microsoft Office 365 People Adapter with Oracle Integration 3</i>
Microsoft Office 365 Outlook Adapter	<i>Using the Microsoft Office 365 Outlook Adapter with Oracle Integration 3</i>

Adapter	For Information
Microsoft SQL Server Adapter	<i>Using the Microsoft SQL Server Adapter with Oracle Integration 3</i>
MLLP Adapter	<i>Using the MLLP Adapter with Oracle Integration 3</i>
MS Dynamics 365 CRM Adapter	<i>Using the MS Dynamics 365 CRM Adapter with Oracle Integration 3</i>
MySQL Adapter	<i>Using the MySQL Adapter with Oracle Integration 3</i>
Netezza Adapter	<i>Using the Netezza Adapter with Oracle Integration 3</i>
OData Adapter	<i>Using the OData Adapter with Oracle Integration 3</i>
Oracle Aconex Cloud Adapter	<i>Using the Oracle Aconex Cloud Adapter with Oracle Integration 3</i>
Oracle Advanced Queuing (AQ) Adapter	<i>Using the Oracle Advanced Queuing (AQ) Adapter with Oracle Integration 3</i>
Oracle Autonomous Data Warehouse Adapter	<i>Using the Oracle Autonomous Data Warehouse Adapter with Oracle Integration 3</i>
Oracle Autonomous Transaction Processing Adapter	<i>Using the Oracle Autonomous Transaction Processing Adapter with Oracle Integration 3</i>
Oracle Cloud Infrastructure Streaming Service Adapter	<i>Using the Oracle Cloud Infrastructure Streaming Service Adapter with Oracle Integration 3</i>
Oracle Commerce Cloud Adapter	<i>Using the Oracle Commerce Cloud Adapter with Oracle Integration 3</i>
Oracle CPQ Adapter	<i>Using the Oracle CPQ Adapter with Oracle Integration 3</i>
Oracle Database Adapter	<i>Using the Oracle Database Adapter with Oracle Integration 3</i>
Oracle Database Cloud Service Adapter	<i>Using the Oracle Database Cloud Service Adapter with Oracle Integration 3</i>
Oracle E-Business Suite Adapter	<i>Using the Oracle E-Business Suite Adapter with Oracle Integration 3</i>
Oracle Eloqua Cloud Adapter	<i>Using the Oracle Eloqua Cloud Adapter with Oracle Integration 3</i>
Oracle CX Sales and B2B Service Adapter	<i>Using the Oracle CX Sales and B2B Service Adapter with Oracle Integration 3</i>
Oracle Enterprise Performance Management Cloud Adapter	<i>Using the Oracle Enterprise Performance Management Cloud Adapter with Oracle Integration 3</i>
Oracle ERP Cloud Adapter	<i>Using the Oracle ERP Cloud Adapter with Oracle Integration 3</i>
Oracle Field Service Cloud Adapter	<i>Using the Oracle Field Service Cloud Adapter with Oracle Integration 3</i>
Oracle HCM Cloud Adapter	<i>Using the Oracle HCM Cloud Adapter with Oracle Integration 3</i>
Oracle Hospitality Adapter	<i>Using the Oracle Hospitality Adapter with Oracle Integration 3</i>
Oracle Intelligent Advisor Adapter	<i>Using the Oracle Intelligent Advisor Adapter with Oracle Integration 3</i>
Oracle Intelligent Track and Trace Adapter	<i>Using the Oracle Intelligent Track and Trace Adapter with Oracle Integration 3</i>
Oracle JD Edwards EnterpriseOne Adapter	<i>Using the Oracle JD Edwards EnterpriseOne Adapter with Oracle Integration 3</i>

Adapter	For Information
Oracle Logistics Adapter	<i>Using the Oracle Logistics Adapter with Oracle Integration 3</i>
Oracle NetSuite Adapter	<i>Using the Oracle NetSuite Adapter with Oracle Integration 3</i>
Oracle Primavera Cloud Adapter	<i>Using the Oracle Primavera Cloud Adapter with Oracle Integration 3</i>
Oracle Primavera Cloud Data Service Adapter	<i>Using the Oracle Primavera Cloud Data Service Adapter with Oracle Integration 3</i>
Oracle Primavera P6 EPPM Adapter	<i>Using the Oracle Primavera P6 EPPM Adapter with Oracle Integration 3</i>
Oracle Responsys Adapter	<i>Using the Oracle Responsys Adapter with Oracle Integration 3</i>
Oracle Service Cloud (RightNow) Adapter	<i>Using the Oracle Service Cloud (RightNow) Adapter with Oracle Integration 3</i>
Oracle Siebel Adapter	<i>Using the Oracle Siebel Adapter with Oracle Integration 3</i>
Oracle SOA Suite Adapter	<i>Using the Oracle SOA Suite Adapter with Oracle Integration 3</i>
Oracle Taleo Enterprise Edition Adapter	<i>Using the Oracle Taleo Enterprise Edition Adapter with Oracle Integration 3</i>
Oracle Unity Adapter	<i>Using the Oracle Unity Adapter with Oracle Integration</i>
Oracle Utilities Adapter	<i>Using the Oracle Utilities Adapter with Oracle Integration 3</i>
Oracle WebLogic JMS Adapter	<i>Using the Oracle WebLogic JMS Adapter with Oracle Integration 3</i>
PayPal Adapter	<i>Using the PayPal Adapter with Oracle Integration 3</i>
PostgreSQL Adapter	<i>Using the PostgreSQL Adapter with Oracle Integration 3</i>
Primavera Unifier Adapter	<i>Using the Oracle Primavera Unifier Adapter with Oracle Integration 3</i>
REST Adapter	<i>Using the REST Adapter with Oracle Integration 3</i>
RosettaNet Adapter	<i>Using the RosettaNet Adapter with Oracle Integration 3</i>
Salesforce Adapter	<i>Using the Salesforce Adapter with Oracle Integration 3</i>
Salesforce Commerce Cloud Adapter	<i>Using the Salesforce Commerce Cloud Adapter with Oracle Integration 3</i>
Salesforce Marketing Cloud Adapter	<i>Using the Salesforce Marketing Cloud Adapter with Oracle Integration 3</i>
Salesforce REST Adapter	<i>Using the Salesforce REST Adapter with Oracle Integration 3</i>
SAP Adapter	<i>Using the SAP Adapter with Oracle Integration 3</i>
SAP ASE (Sybase) Adapter	<i>Using the SAP ASE (Sybase) Adapter with Oracle Integration 3</i>
SAP Ariba Adapter	<i>Using the SAP Ariba Adapter with Oracle Integration 3</i>
SAP Commerce Cloud (Hybris) Adapter	<i>Using the SAP Commerce Cloud (Hybris) Adapter with Oracle Integration 3</i>
SAP Concur Adapter	<i>Using the SAP Concur Adapter with Oracle Integration 3</i>

Adapter	For Information
SAP SuccessFactors Adapter	<i>Using the SAP SuccessFactors Adapter with Oracle Integration 3</i>
SAP S/4HANA Cloud Adapter	<i>Using the SAP S/4HANA Cloud Adapter with Oracle Integration 3</i>
ServiceNow Adapter	<i>Using the ServiceNow Adapter with Oracle Integration 3</i>
Shopify Adapter	<i>Using the Shopify Adapter with Oracle Integration 3</i>
Shopify GraphQL Adapter	<i>Using the Shopify GraphQL Adapter with Oracle Integration 3</i>
Slack Adapter	<i>Using the Slack Adapter with Oracle Integration 3</i>
Snowflake Adapter	<i>Using the Snowflake Adapter with Oracle Integration 3</i>
SOAP Adapter	<i>Using the SOAP Adapter with Oracle Integration 3</i>
SugarCRM Adapter	<i>Using the SugarCRM Adapter with Oracle Integration 3</i>
SurveyMonkey Adapter	<i>Using the SurveyMonkey Adapter with Oracle Integration 3</i>
Trello Adapter	<i>Using the Trello Adapter with Oracle Integration 3</i>
Twilio Adapter	<i>Using the Twilio Adapter with Oracle Integration 3</i>
Twitter Adapter	<i>Using the Twitter Adapter with Oracle Integration 3</i>
Workday Adapter	<i>Using the Workday Adapter with Oracle Integration 3</i>
Zendesk Adapter	<i>Using the Zendesk Adapter with Oracle Integration 3</i>
Zuora Adapter	<i>Using the Zuora Adapter with Oracle Integration 3</i>

3. Click **Save**.

For most adapters, you are now ready to test your connection. For some adapters, you must first configure an agent group. See [Create a Connection with an Agent Group](#).

Test the Connection

Test your connection to ensure that it's configured successfully.

- In the page title bar, click **Test**. What happens next depends on whether your adapter connection uses a Web Services Description Language (WSDL) file. Only some adapter connections use WSDLs.

If Your Connection...	Then...
Doesn't use a WSDL	The test starts automatically and validates the inputs you provided for the connection.

If Your Connection...	Then...
Uses a WSDL	A dialog prompts you to select the type of connection testing to perform: <ul style="list-style-type: none">• Validate and Test: Performs a full validation of the WSDL, including processing of the imported schemas and WSDLs. Complete validation can take several minutes depending on the number of imported schemas and WSDLs. No requests are sent to the operations exposed in the WSDL.• Test: Connects to the WSDL URL and performs a syntax check on the WSDL. No requests are sent to the operations exposed in the WSDL.
	2. Wait for a message about the results of the connection test. <ul style="list-style-type: none">• If the test was successful, then the connection is configured properly.• If the test failed, then edit the configuration details you entered. Check for typos and verify URLs and credentials. Continue to test until the connection is successful.
	3. When complete, click Save .

Edit a Connection

You can edit connection settings after creating a new connection.

- 1.** In the navigation pane, click **Design**, then **Connections**.
- 2.** Hover your cursor over the connection to edit.
- 3.** Click **Edit**  or click the connection name.

 **Note:**

You can also edit a connection from the integration canvas. See [Define Inbound Triggers and Outbound Invokes](#).

- 4.** On the Connection page, make any necessary edits.

If you edit a connection currently used by an active integration, you can click **Save & Reactivate** to apply the connection changes to active integrations. Schedule integrations with an active schedule are excluded. Any queued or in-progress schedule integration instances are aborted.

Clone a Connection

You can clone a copy of an existing connection, even if the connection is locked. This provides a quick way to create a new connection.

- 1.** In the navigation pane, click **Design**, then **Connections**.
- 2.** Hover over the connection to clone.
- 3.** Click **Actions ...**, then select **Clone**.
The Clone connection dialog is displayed.
- 4.** Enter any necessary connection information.
- 5.** Click **Clone**.

Cloning a connection does not copy the credentials.

6. Hover your cursor over the cloned connection, then click **Edit**  to configure the credentials of your cloned connection.

See [Edit a Connection](#).

Delete a Connection

You can delete a connection that is no longer needed.

1. In the navigation pane, click **Design**, then **Connections**.
2. Hover over the connection to delete.
3. Click **Actions** , then select **Delete**.

If the connection is currently used in an active integration, it cannot be deleted. Deleting the connection would invalidate the integration. Active integrations would stop running. Remove the connection from the integration and then delete the connection.

4. Click **Delete** to confirm deletion.

Unlock a Connection

When a connection is in edit mode and the browser fails, the connection becomes locked, which prevents it from being edited.

Unlock the connection in either of two ways:

1. Log in again as the same user who was editing the connection when the browser failed.
For example, if you were logged in as `oicdeveloper` when the browser failed, log in again as `oicdeveloper`.
2. Log out.
This action unlocks the connection.

or

1. Log in as a user with the Administrator role.
2. In the navigation pane, click **Design**, then **Connections**.
3. Hover over the connection that is locked.
4. Click **Actions** , then select **Unlock**.
5. Click **Unlock** when prompted to confirm.

Refresh 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.

 **Note:**

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

1. In the navigation pane, click **Design**, then **Connections**.
2. Hover over the connection to refresh.
3. Click **Actions . . .**, then select **Refresh metadata**.

A message is displayed indicating that the refresh was successful.

Manage Security Certificates

You can manage security certificates in Oracle Integration.

Topics:

- [Upload a Certificate to Connect with External Services](#)
- [Update or Delete a Certificate](#)

Upload a Certificate to Connect with External Services

Certificates allow Oracle Integration to connect with external services. If the external service/endpoint needs a specific certificate, request the certificate and then import it into Oracle Integration.

If you make an SSL connection in which the root certificate does not exist in Oracle Integration, an exception error is thrown. In that case, you must upload the appropriate certificate. A certificate enables Oracle Integration to connect with external services. If the external endpoint requires a specific certificate, request the certificate and then upload it into Oracle Integration.

1. Sign in to Oracle Integration.
2. In the navigation pane, click **Settings**, then **Certificates**.
All certificates currently uploaded to the trust store are displayed on the Certificates page.
3. Click **Filter**  to filter by name, certificate expiration date, status, type, category, and installation method (user-installed or system-installed). Certificates installed by the system cannot be deleted.

Certificates			
Name	Type	Category	Status
akt_pgppublic	PGP	Public	Configured
akt_pgpprivate	PGP	Private	Configured
testpgppublic	PGP	Public	Configured
testpgppsecret	PGP	Private	Configured
elq_cert1	X.509	Trust	Configured
Egir_CloudCA	SAML	Message Protection	Configured
qa_lan	X.509	Trust	Configured
OpportunityServiceSoapHttpPort	X.509	Trust	Configured
DigiCertCA2	X.509	Trust	Configured
SG-Utilities	X.509	Trust	Configured
app_elq_p01	X.509	Trust	Configured

4. Click **Upload** at the top of the page.
The Upload certificate panel is displayed.
5. Enter an alias name and optional description.
6. In the **Type** field, select the certificate type. Each certificate type enables Oracle Integration to connect with external services.
 - [Digital Signature](#)
 - [X.509 \(SSL transport\)](#)
 - [SAML \(Authentication & Authorization\)](#)
 - [PGP \(Encryption & Decryption\)](#)
 - [Signing key](#)

Digital Signature

The digital signature security type is typically used with adapters created with the Rapid Adapter Builder. See Learn About the Rapid Adapter Builder in Oracle Integration in *Using the Rapid Adapter Builder with Oracle Integration 3*.

1. Click **Browse** to select the digital certificate. The certificate must be an X509Certificate. This certificate provides inbound RSA signature validation. See RSA Signature Validation in *Using the Rapid Adapter Builder with Oracle Integration 3*.
2. Click **Upload**.

X.509 (SSL transport)

1. Select a certificate category.
 - a. **Trust:** Use this option to upload a trust certificate.
 - i. Click **Browse**, then select the trust file (for example, .cer or .crt) to upload.
 - b. **Identity:** Use this option to upload a certificate for two-way SSL communication.
 - i. Click **Browse**, then select the keystore file (.jks) to upload.
 - ii. Enter the comma-separated list of passwords corresponding to key aliases.

 **Note:**

When an identity certificate file (.jks) contains more than one private key, all the private keys must have the same password. If the private keys are protected with different passwords, the private keys cannot be extracted from the keystore.

- iii. Enter the password of the keystore being imported.
- c. Click **Upload**.

SAML (Authentication & Authorization)

1. Note that **Message Protection** is automatically selected as the only available certificate category and cannot be deselected. Use this option to upload a keystore certificate with SAML token support. Create, read, update, and delete (CRUD) operations are supported with this type of certificate.
2. Click **Browse**, then select the certificate file (.cer or .crt) to upload.
3. Click **Upload**.

PGP (Encryption & Decryption)

1. Select a certificate category. Pretty Good Privacy (PGP) provides cryptographic privacy and authentication for communication. PGP is used for signing, encrypting, and decrypting files. You can select the private key to use for encryption or decryption when configuring the stage file action.

See olink:ICSUG-GUID-4357B1C8-0FBB-482F-9ED4-041ADE1B857F.

- a. **Private:** Uses a private key of the target location to decrypt the file.
 - i. Click **Browse**, then select the PGP file to upload.
 - ii. Enter the PGP private key password.
- b. **Public:** Uses a public key of the target location to encrypt the file.
 - i. Click **Browse**, then select the PGP file to upload.
 - ii. In the **ASCII-Armor Encryption Format** field, select **Yes** or **No**.
 - **Yes** shows the format of the encrypted message in ASCII armor. ASCII armor is a binary-to-textual encoding converter. ASCII armor formats encrypted messaging in ASCII. This enables messages to be sent in a standard messaging format. This selection impacts the visibility of message content.
 - **No** causes the message to be sent in binary format.
 - iii. From the **Cipher Algorithm** list, select the algorithm to use. Symmetric-key algorithms for cryptography use the same cryptographic keys for both encryption of plain text and decryption of cipher text. The following supported cipher algorithms are FIPS-compliant:
 - AES128
 - AES192
 - AES256
 - TDES

- c. Click **Upload**.

Signing key

A signing key is a secret key used to establish trust between applications. Signing keys are used to sign ID tokens, access tokens, SAML assertions, and more. Using a private signing key, the token is digitally signed and the server verifies the authenticity of the token by using a public signing key. You must upload a signing key to use the OAuth Client Credentials using JWT Client Assertion and OAuth using JWT User Assertion security policies in REST Adapter invoke connections. Only PKCS1- and PKCS8-formatted files are supported.

1. Select **Public or Private**.
2. Click **Browse** to upload a key file.
If you selected **Private**, and the private key is encrypted, a field for entering the private signing key password is displayed after key upload is complete.
3. Enter the private signing key password. If the private signing key is not encrypted, you are not required to enter a password.
4. Click **Upload**.

Update or Delete a Certificate

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

1. In the navigation pane, click **Settings**, then **Certificates**.
2. Use search or filter the list to find the certificate you want to update or delete.
3. Hover over the certificate name, and click **Actions ...**.
4. To update the certificate, select **Update**.
 - a. Update the certificate as required, such as uploading a new certificate. See [Upload a Certificate to Connect with External Services](#).

 **Note:**

You cannot update an identity certificate. To change an identity certificate, you must first delete and then re-upload it.

- b. Click **Update**.
5. To delete the certificate, select **Delete**, and then click **Delete** when prompted to confirm.

View Preinstalled Adapters

You can view the adapters preinstalled with Oracle Integration. You can also create a connection for a selected adapter on this page.

1. In the navigation pane, click **Design**, then **Adapters**.

Name	Version	Type
AQ	19.3.5	Preinstalled
Adobe Sign	19.4.3	Preinstalled
Apache Kafka	20.3.2	Preinstalled
Ariba	21.2.5	Preinstalled
Box	19.4.1	Preinstalled
Concur	21.3.2	Preinstalled
DB2	20.1.4	Preinstalled
DocuSign	21.2.1	Preinstalled
Eloqua	21.1.2	Preinstalled
Eventbrite	19.4.1	Preinstalled

2. Click the adapter name or hover over a row and click **View**  to display information about the adapter.
3. To create an adapter connection, hover over its name, then click **Actions**  and select **Create Connection**. The Create connection panel is displayed for that adapter.

Allow JSON Numbers with High Precision and Scale

For JSON numbers with high precision and scale, Oracle Integration automatically converts the number to four decimal places. To prevent this conversion, you have several options based on the adapter you are using.

- For the REST Adapter, you must set the **Allow High Precision Numbers** connectivity property in the mapper to allow JSON numbers with high precision and scale. The default value for this connectivity property is false. Oracle Integration internally restricts the precision scale to four (decimal places) by default. Set this property to true for a higher precision scale. Use this property value carefully because undefined precision does not mean infinite precision.

Mapping canvas	Target
	Request (REST) *
	Connectivity Properties *
	Rest API
	Plugin *
	Post Query String
	Use Form URL Encoding
	Enforce Empty JSON Object Payload
	Enforce Absolute Endpoint URI
	Skip Control Characters
	Allow High Precision Numbers

- For all other adapters that supply a schema for response processing, the following schema annotation is included.

```
<xsd:schema xmlns:nxsd="http://xmlns.oracle.com/pcbpel/nxsd"
nxsd:version="JSON" nsdx:allowHighPrecisionNumbers="true">
```

If you receive a REST response with more than 10 decimal places, Oracle Integration converts it to XML, but does not round off the values.

If you do not use this schema annotation, the value is rounded off at four decimal places.

5

Create Application Integrations

You can create business object- or business event-based, application integrations. Application integrations can be request/reply or fire-and-forget types. Application integrations use Oracle BPEL Process Manager capabilities. Oracle BPEL Process Manager enables you to define how a business process that involves web services runs. BPEL messages invoke remote services and orchestrate the running of processes. When designing integrations, you can create multiple routing expressions.

Note:

Ensure that you optimize integration design. If an application integration contains too many actions (for example, there are 25 or more stage write actions), activation failure can occur. For example, split an integration with many actions into multiple integrations. See [Troubleshoot Integration Activations](#) and [Design Best Practices](#).

Topics:

- [Create an Integration](#)
- [Design an Application Integration](#)
- [Define Inbound Triggers and Outbound Invokes](#)
- [Add Suggestions, Invokes, and Actions Through an Inline Menu](#)
- [Cut and Paste Elements in the Integration Canvas](#)
- [Expand and Collapse Actions in the Integration Canvas](#)
- [Create Global Variables](#)
- [Define Ad-Hoc Maps](#)
- [Import a Map File into an Integration](#)
- [Add Actions to an Application Integration](#)
- [Use XPath Axis and Wildcard Expressions in the Expression Builder](#)
- [Use Lookups in Variable Assignments](#)
- [Define Fault Aggregation in Parallel Processing Scenarios](#)
- [Assign Business Identifiers for Tracking Fields in Messages](#)
- [Display Errors and Warnings in an Integration](#)

Create an Integration

Creating an integration includes defining the trigger and invoke application connections, and defining how data is mapped between the two applications. The following procedure provides general instructions for creating an integration, with links to more detailed information for

certain steps. When 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 [Manage Lookups](#) for instructions.

To create an integration:

1. In the navigation pane, click **Design**, then **Integrations**.

2. On the Integrations page, click **Create**.

The Create integration panel opens.

3. Select the type of integration pattern applicable to your business needs. See [Understand Integration Patterns](#).

4. Enter the following information:

Field	Description
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: <ul style="list-style-type: none">• INT-A/2.00.0000 - Not active• INT-A/2.10.0000 - Not active
	Integration INT-A/2.00.0000 is then activated. <ul style="list-style-type: none">• INT-A/2.00.0000 is now active.• INT-A/2.10.0000 is not active.
	Integration 2.10.0000 is then activated. <ul style="list-style-type: none">• INT-A/2.00.0000 is now not active.• INT-A/2.10.0000 is now active.

Field	Description
Documentation URL	Optionally enter a URL to a design document that provides the details of the integration. The valid URL format is: <code>http(s)://hostname:port/ documentation_path</code>
Keywords	Enter keywords (tags) to identify the integration. You can search for integrations on the Integrations page using keywords.
Package	Enter a new or existing package name in which to place your integration. When you enter the initial letters of an existing package, it is displayed for selection. See Manage Packages and About Packages .
Description	Provide a meaningful description so that others can understand the integration.
Available to other projects	Selecting this check box enables this project integration to be called by a parent integration in a different project. If you don't select this check box, this integration can only be called by a parent integration in the same project. See Invoke a Child Integration from a Parent Integration .

5. Click **Create**.
6. See the following sections based on your selection.

If You Selected...	See...
Application	Design an Application Integration
Schedule	Create a Schedule Integration
Event	Subscribe to Events in an Integration

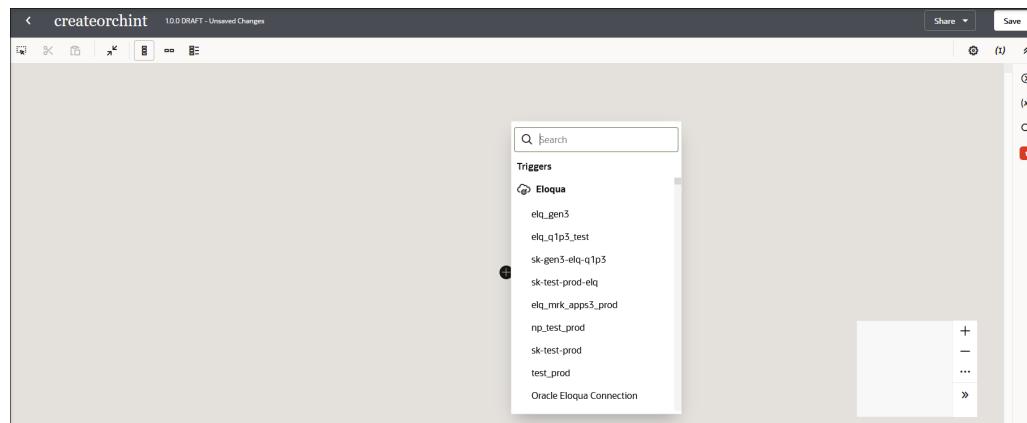
Design an Application Integration

This section describes how to use the integration canvas to design an application integration. It also provides an overview of the integration canvas in which you design your integration.

1. Follow the steps in [Create an Integration](#) to create an application integration.

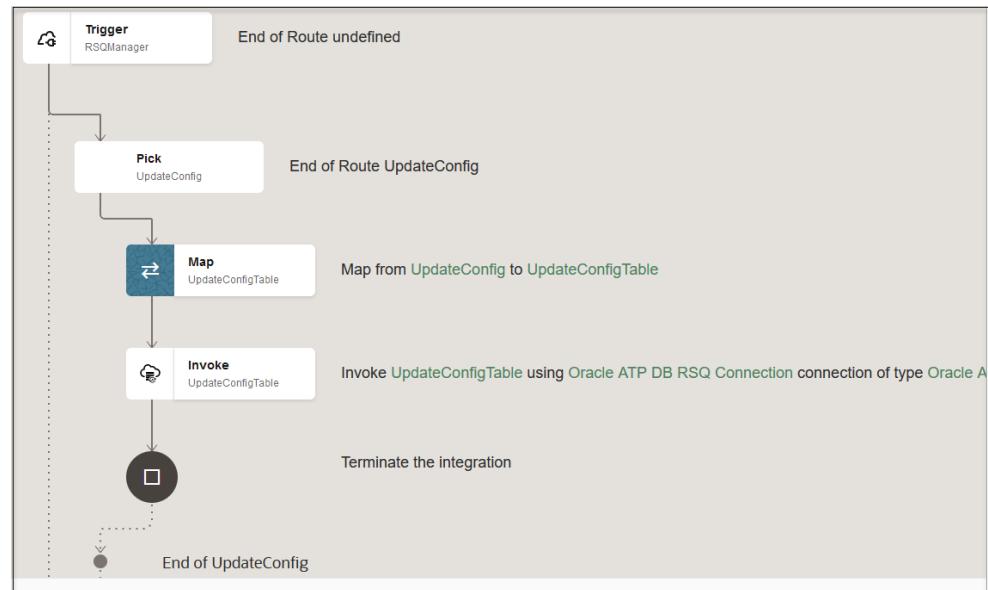
An empty integration canvas with the following sections is displayed:

- The empty integration canvas is identified by a + sign and a **Triggers** menu of available trigger adapter connections in the middle of the canvas. The **Triggers** menu is automatically displayed the first time. After that, you need to click the + sign to invoke an inline menu for designing your integration. Click an adapter to display the number of configured adapter connections available for adding to an integration. This invokes the Adapter Endpoint Configuration Wizard to create an inbound connection in the integration.



- You can also select **Triggers** in the right pane to display the same menu of available trigger adapter connections.
- Click the following items to design the integration:

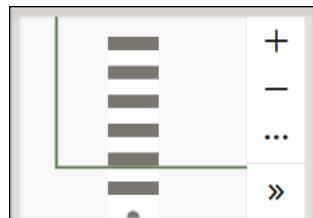
- **Selection Mode** : Select elements in the canvas for cutting and pasting.
- **Cut** : Cut actions and invokes to paste elsewhere in the integration.
- **Paste** : Paste actions and invokes to other sections of the integration.
- **Collapse All Collections** : Collapse logic actions such as a scope, for each, switch, while, and others that include child actions.
- **Refresh Endpoints** : Click to refresh endpoints in the integration. This icon appears if your integration includes connections that can be regenerated. See [Refresh Endpoints for Integrations](#).
- **Vertical** : Display a vertical view of the integration.
- **Horizontal** : Display a horizontal view of the integration.
- **Outline** : Display the integration vertically with child actions indented. Details about each action in the integration are displayed to the right. You can edit the integration by selecting the + sign to display an inline menu for adding invokes and actions. You can also edit actions or add routes to switches by clicking the appropriate action.



- ▷/□: Start and end integration testing from within the integration canvas. These icons only appear for REST Adapter connection trigger-based integrations and scheduled integrations. See [Test Integrations from Inside the Integration Canvas](#).
- **Share** drop-down list: Export the integration or print the design of the integration.
- **Primary Info** : Display a panel showing integration creation properties (identifier, name, version, package, documentation URL, keywords, smart tags, and description), a button to delete the integration, and a report that shows details about the actions and invokes used in the integration. You can change the name, package, documentation URL, and description in this panel.
- **Business Identifiers** : Create business identifiers for the integration. At least one identifier is required. Business identifiers enable you to track integration instances during runtime. See [Track Integration Instances](#).
- **Minimize** and **Maximize** : Minimize to show the options above the integration canvas and maximize to hide the options above the integration canvas.
- **View Global Fault Handler** : View the global fault action and re-throw fault action. You can add and delete re-throw faults and other actions, as necessary.
- **Triggers** : Create inbound triggers for the integration. This icon does not appear after you create the trigger for your integration.
- **Invokes** : Create outbound invokes for the integration.
- **Actions** : Display a pane of action that you can add to the integration. See [Define Inbound Triggers and Outbound Invokes](#).
- **Global Variables** : Add a global variable to the integration that can be selected for use in actions.

- **Search**  : Search for triggers, invokes, and actions in the integration. You can select the element in the drop-down list or the structure.

At the bottom right of the integration canvas is a small box with icons for adjusting the integration view, including zooming in, zooming out, expanding the integration to 100%, fitting the integration into the canvas, and minimizing/maximizing the box.

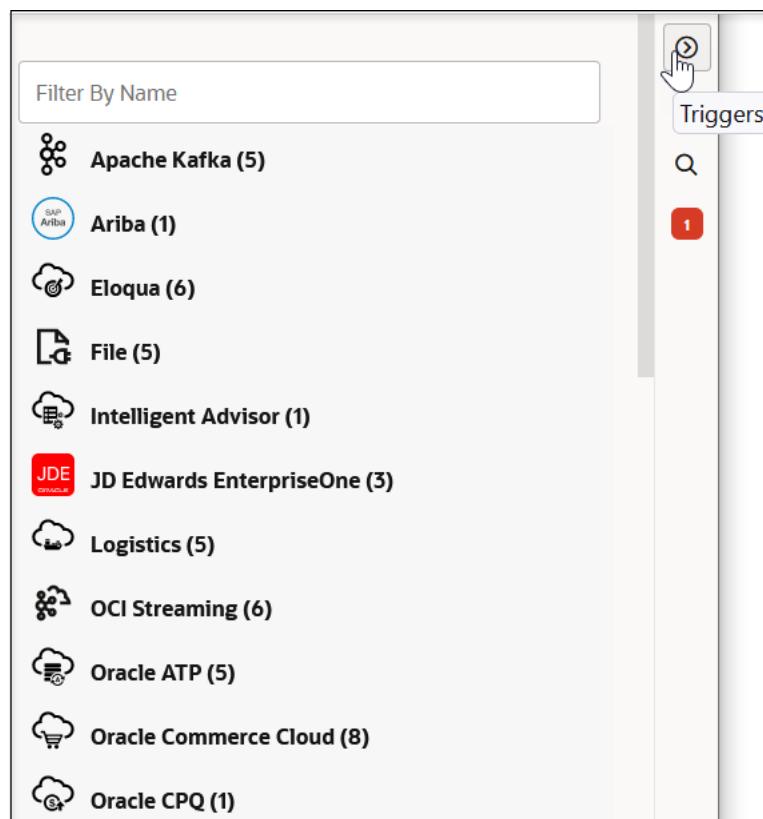


Define Inbound Triggers and Outbound Invokes

You define a single inbound trigger and one or more outbound invokes in your integration.

1. To add a trigger connection to the integration canvas, click + in the integration canvas or click **Triggers**  on the side of the canvas. Both options show a list of available trigger connections. This icon disappears once you define the trigger.

If the adapter connection you want to use does not exist, click **Triggers**  then + to create a new connection.



The Create connection panel is displayed for creating the connection. Once the adapter connection is configured, tested, and the message indicates that it is 100% complete, save and close the page. The new adapter connection is displayed in the panel. If you edit a connection so that it is now *not* 100% complete, then save and close the Connections page, a message is displayed in the banner indicating that it is not 100% complete and cannot be displayed in the panel. For the connection to be displayed in the panel again, you must return to the Connections page and configure the connection to be 100% complete.

2. Add the configured adapter by clicking + in the integration canvas and selecting it from the list.

The Adapter Endpoint Configuration Wizard is displayed.

3. Complete the pages of the wizard to configure the selected adapter. See [Understand Trigger and Invoke Connections](#).

When complete, a configured trigger is displayed in the canvas. An unconfigured mapper icon is displayed immediately after the trigger. Because this trigger was configured to send a response in this example, a return icon is displayed in the integration canvas. You *cannot* delete a trigger in an integration; no delete option is available when you click the trigger and select **Actions ...**.

 **Note:**

On the side of the canvas, the **Triggers**  icon is replaced by **Invokes**  icon, which enables you to add multiple outbound invoke connections to the integration.

An **Actions**  icon is now displayed after **Invokes** . When expanded, this section displays the following actions that you can add to the integration:

Actions:

- **Assign:** Assign variables and values using the Expression Builder.
- **B2B:** Translate a message to or from the Electronic Data Interchange (EDI) format in the integration. This action converts an EDI document into an Oracle Integration message, or vice versa.
- **Data stitch:** Incrementally build a message payload from one or more existing payloads. The stitch action provides an editor for assigning values to variables.
- **Logger:** Log messages to the activity stream and logs by adding the logger action at any point in the integration. You can create a log message that is a static message or variable-populated message in the Expression Builder.
- **Map:** Add mappings to the integration, when needed. Mappings are also automatically added when you add triggers, invokes, and some actions to an integration. In most cases, the messages you transfer between applications in an integration have different data structures. Mappings enable you to define how data is transferred, or mapped, between applications.
- **Note:** Add placeholder notes similar to sticky notes to an integration. For example, you have not yet defined an invoke connection and want to add a placeholder note in the integration indicating that you plan to define the invoke connection later.

- **Notification:** Send a notification email to relevant users at specific points in the processing of an integration. You can set the To, From, and Subject parts of an email. You can also create the body part of an email using parameters defined in the Expression Builder.
- **Stage File:** Process files in integrations. The stage file action can process each file downloaded by the FTP Adapter. The stage file action can read (and remove any trailer), write, zip, and unzip files in a stage integration known to Oracle Integration.
- **Wait:** Delay the processing of an integration for a specified period of time. A typical use case for this action is to invoke a specific operation at a certain time.

Error Handling:

- **Re-throw Fault:** Send failed messages to the error hospital for further analysis. If the integration contains a defined global fault, the error captured by the re-throw fault action is sent through the global fault and onto the error hospital for analysis. If no global fault is defined, the fault is sent.
- **Throw New Fault:** Generate an error and configure a code, reason, details, and skip condition for that error at a certain point during the processing of an integration.

Logic:

- **For each:** Loop over a repeating element and process one or more actions within the scope of the for-each action. The number of looping iterations is based on a user-selected repeating element.
- **Parallel:** Process tasks in parallel to improve integration performance and response times. A parallel action allows the path of an integration to be split into multiple branches.
- **Scope:** Manage a group of actions. The scope action is a collection of child actions and invokes that can have their own fault handlers. The scope action provides the behavior context for the child actions. The actions defined in the parent scope have local visibility inside the scope.
- **Switch:** Define branches to add routing expressions in the integration. You define one or more branches. An otherwise branch is taken if the routing expressions for the initial branches do not resolve to true.
- **While:** Loop over actions or invoke connections while a specific condition is met. You define the condition for the while loop in the Expression Builder.

Call:

- **File server:** Obtain a list of files, write or move a file to File Server, download and read a file from anywhere, and delete a file on File Server.
- **Integration:** Invoke another integration from an integration. When you drag this action into the integration, you are prompted to select an active integration to invoke.
- **Javascript:** Add JavaScript functions to the integration. This enables you to call JavaScript functions from an integration.
- **OCI Function:** Invoke native Oracle Cloud Infrastructure functions.
- **OCI Object storage:** Invoke Oracle Cloud Infrastructure Object Storage.
- **Publish event:** Publish an event in an integration to which subscription integrations can subscribe. See [About Publishing and Subscribing to Events](#).

End:

- **Callback:** End a process and return to the trigger. For example, you can add a switch activity and define a branch in which you add a **Callback**. If some defined expression

logic is not met, this branch is taken. The integration is stopped and the trigger receives a response indicating that the integration is not continuing.

- **Fault return:** Return a fault. If the trigger is synchronous with at least one fault, the fault return action is available. If a trigger defines multiple faults, you are prompted to select a fault. If the trigger is asynchronous with at least one fault, the fault return action is available. Once the fault is added, a mapper with the type of fault definition is also added. This action is valid only if the trigger has at least one fault.
- **Return:** Return an immediate response.
- **Stop:** Terminate the integration. No response message is returned to the trigger.

4. On the side of the canvas, click **Invokes**  to expand the panel. As with the trigger connection, you can click the + sign to create a new invoke connection. See [Step 1](#) for information.
5. Click the adapter type (for example, a REST Adapter) to display the configured connections for that adapter.
6. Drag the specific configured adapter connection to the + sign in the integration canvas.
7. Drop the adapter.

This invokes the Adapter Endpoint Configuration Wizard.

8. Complete the pages of the wizard to configure the invoke connection.

When complete, a configured invoke connection is displayed in the integration canvas.

You can click the trigger and invoke connections to edit or view their contents. If you re-edit the selected connection in the Adapter Endpoint Configuration Wizard and click **Done** (even without making any changes) you are prompted to update the configuration and regenerate the artifacts.

- If you select **Yes** after making only minor changes, the system validates the flow and displays the warnings and errors (if there are any) so you can fix any potential problems.
- If you select **Yes** after making major changes to the trigger connection (for example, changing the message exchange pattern of the trigger from synchronous to asynchronous), all mappings, replies, and stop elements are deleted from the integration except for the system-generated reply or stop element at the end of the integration. Tracking information is also deleted.

Editing an invoke connection can also result in major change. However, the impact is minimal compared to editing a trigger. Invoke editing does not have flow-wide consequences such as maps and other actions being removed.

You can place your cursor anywhere in the canvas to move the integration, when needed. You can also move the integration from within the large box in the upper right corner.

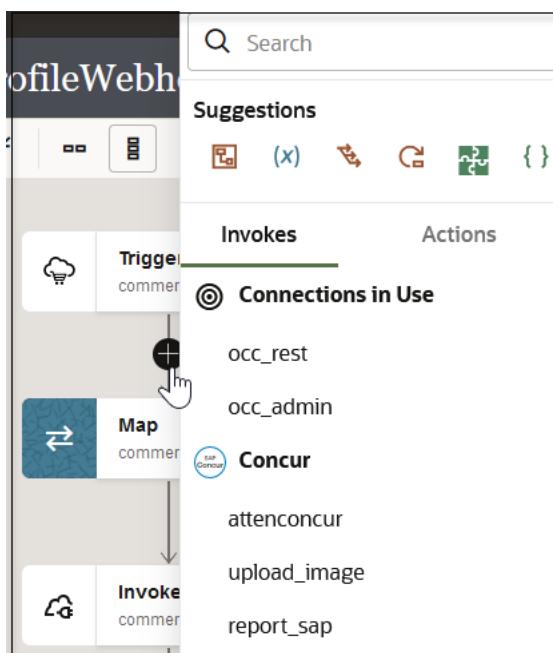
Add Suggestions, Invokes, and Actions Through an Inline Menu

As an alternative to dragging suggestions, invoke connections, and actions from the right navigation pane, you can hover your cursor over the + sign that is displayed between the elements in your integration. When you click the + sign, a list of actions and invokes available for adding to your integration is displayed. A search facility is also available.

1. Hover over the lines between the elements to display a + sign.
2. Click the + sign.



A menu with suggestions, invokes, and actions is displayed.



Element	Description
Suggestions	Displays actions that are commonly placed between the two elements where you clicked.
Invokes	Displays adapter invoke connections available to add. The option is only displayed when a connection can be added after the previous element.

Element	Description
Actions	Displays actions that are allowed to be placed between the two elements where you clicked.

3. Click the **Invokes** tab or **Actions** tab.
4. Select the appropriate invoke or action to add to your integration or begin entering the action type (for example, a while action) or invoke type (for example, a SOAP Adapter) in the **Search** field to show matching elements to add.

The dialog box or wizard specific to your selection is displayed for configuration. For example, if you add a connection, the Adapter Endpoint Configuration Wizard is displayed.

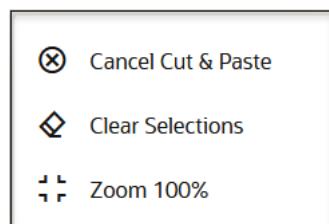
Cut and Paste Elements in the Integration Canvas

You can cut and paste elements (actions and invokes) in the integration canvas. This enables you to move elements to other places in your integration and eliminates the need to recreate them. The cut and paste feature replaces the reposition feature in Oracle Integration Generation 2.

- [Guidelines](#)
- [Options for Cutting and Pasting Elements](#)
- [Example: Cut and Paste a Single Element](#)
- [Example: Cut and Paste Multiple Elements](#)

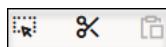
Guidelines

- You can cut and paste individual elements or multiple contiguous elements.
- You can cut and paste more logically complex actions such as a scope action and all the elements within its container or branching actions such as a switch action that includes multiple branching actions such as for-each actions inside it.
- You cannot cut a trigger, either individually or as part of a group of selected elements.
- You cannot cut an end node.
- You cannot cut branches from some actions, such as a routing branch in a switch action.
- If you cut an element and then decide you don't want to paste it, press the **Esc** key or right-click anywhere in the integration canvas and select **Cancel Cut & Paste**. This cancels the cut action.



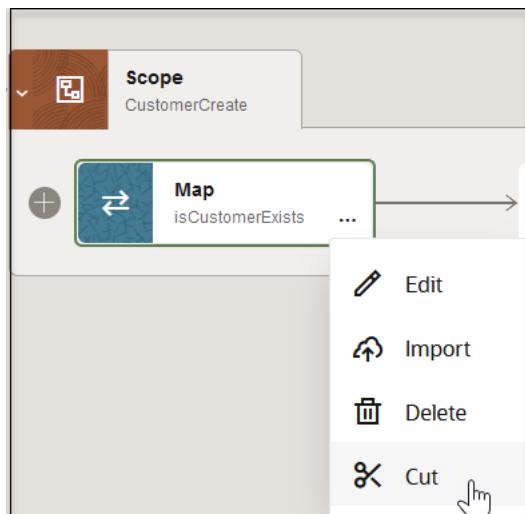
Options for Cutting and Pasting Elements

You can cut and paste elements in the integration canvas with toolbar options above the integration canvas or with inline menus.



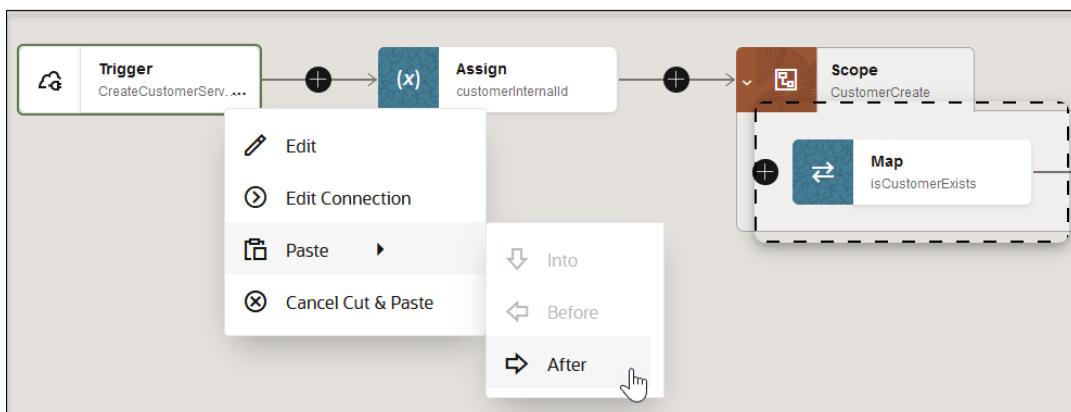
You can cut elements in multiple ways:

- Select a single element or multiple elements (with the **Selection Mode** option) in the integration canvas and click in the toolbar.
- Select a single element or multiple elements in the integration canvas, then click **Actions** and select **Cut** for the selected element or elements.

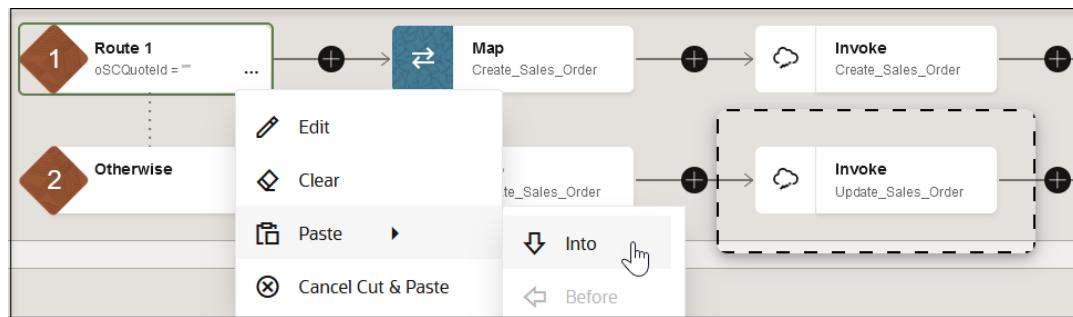


You can paste elements in multiple ways:

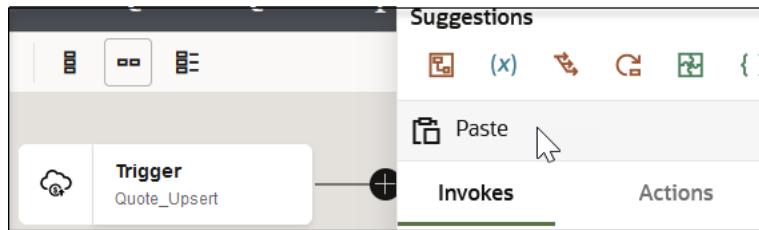
- Select the element either before or after the location at which you want to paste it, then click **Actions** and select **Paste**, and either **Before** or **After** (depending on which element you selected).



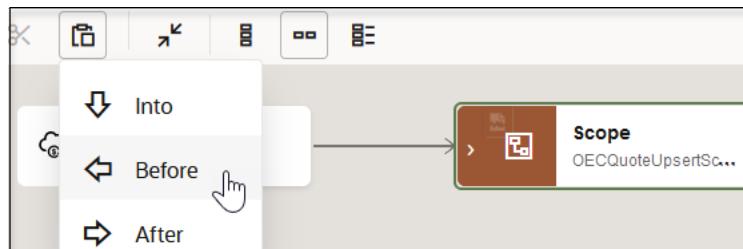
- Paste inside an action such as a scope or a branch of a switch action by clicking **Actions** and selecting **Paste**, and then **Into**.



- Click **Insert** in the location at which to move the element, then select **Paste**.



- Click **Paste** above the toolbar to paste the selected element in the appropriate location. The selections you make in the integration canvas determine when the toolbar options are enabled.



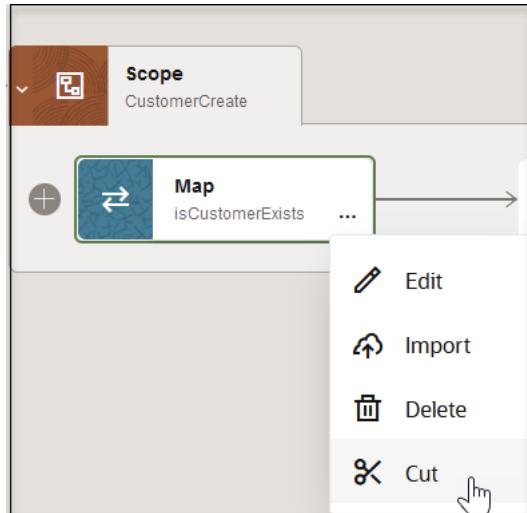
Example: Cut and Paste a Single Element

This example shows one way to cut and paste a single element. Other cut and paste methods are described in [Options for Cutting and Pasting Elements](#).

- Click the element to move. For this example, a map action is selected to move. This action enables **Cut** in the toolbar.



2. In the selected action, click **Actions ...** and select **Cut**.

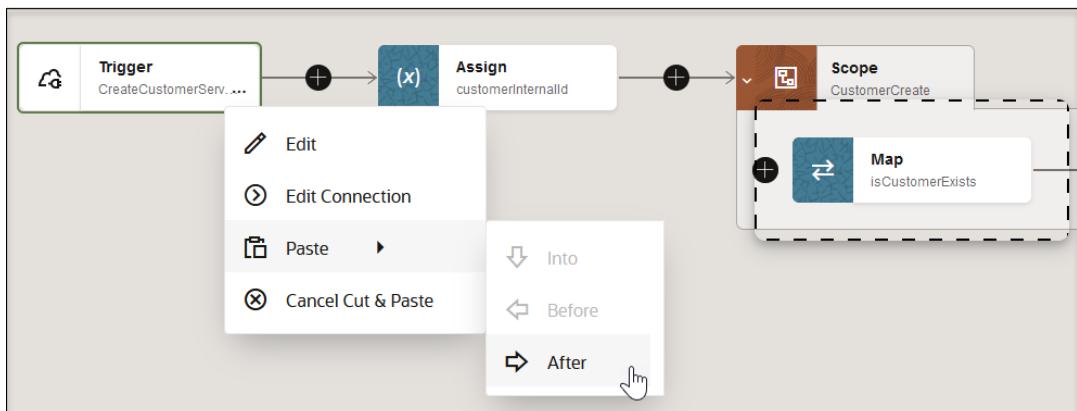


An animated dashed box is displayed around the action to cut. You cannot click inside this box and perform any actions. You can also expand the number of elements to cut by dragging the dashed box around other elements.

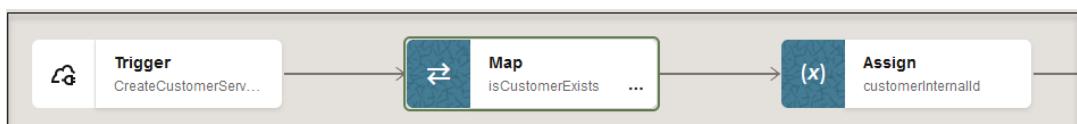
3. Select the element next to where you want to paste the element.

4. Click **Actions ...** and select **Paste**, then **After**.

For this example, the trigger element is selected. As an alternative, you can also select the assign action and paste the map action before it. The menu context changes based on where you select to paste the element.



The map element is pasted to its new location after the trigger and before the assign action.

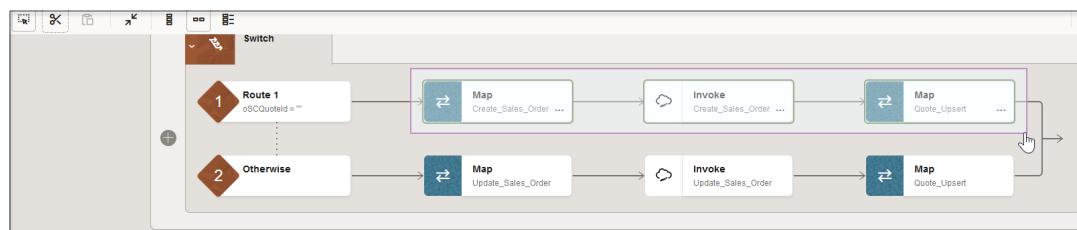


Example: Cut and Paste Multiple Elements

This example shows one way to use cut and paste multiple elements. Other cut and paste methods are described in [Options for Cutting and Pasting Elements](#).

1. Click **Selection Mode**  in the toolbar.

2. Drag your cursor around the elements to cut. For this example, a map, an invoke, and another map are selected.



3. Click **Cut**  in the toolbar.

An animated dashed box is displayed around the elements to cut. You cannot click inside this box and perform any actions.

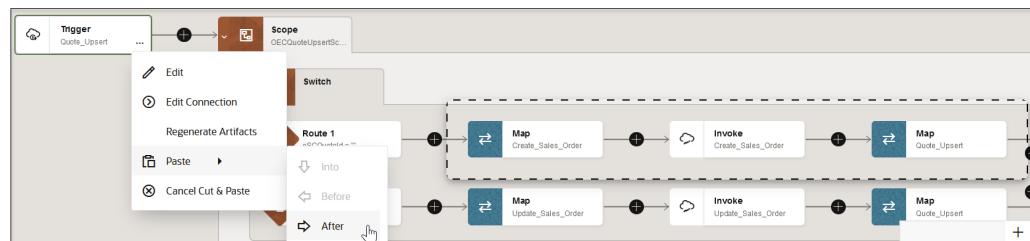


4. Select the element next to where you want to paste the element. For this example, the trigger is selected.

5. Paste the selected elements after the trigger in either of two ways:

- a. Click **Actions**  and select **Paste**, then **After**.
or

- b. Click **Paste**  in the toolbar, then select **After**.



The map, invoke, and map are pasted to their new location after the trigger and before the scope action.

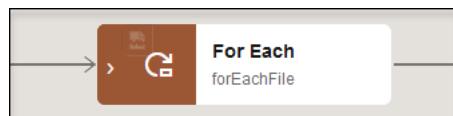


Expand and Collapse Actions in the Integration Canvas

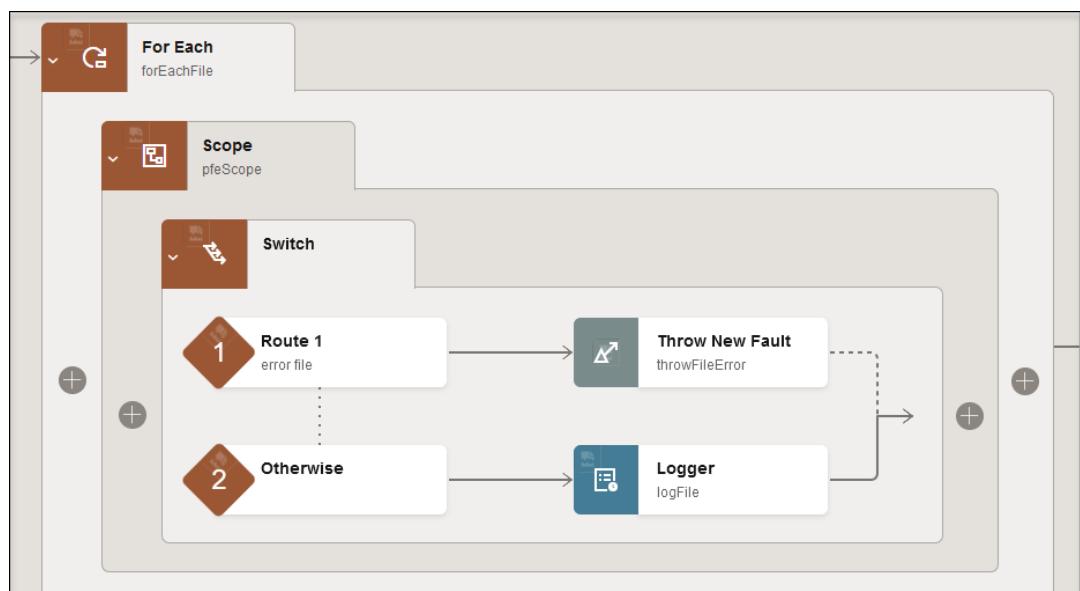
You can expand and collapse actions that include other actions and invokes in the integration canvas. This includes logic actions such as a scope, for-each, while, and switch; stage file actions using the read file in segments operation; and parent integrations that invoke child integrations.

The following methods enable you to toggle back and forth between expanded and collapsed actions:

- Double-click an action.
 - Click the **expand (v)** and **collapse (>)** icons on an action.
 - Click **Actions ...** and select **Expand or Collapse**.
1. Expand the action (for this example, a for-each action) through one of the methods described above.



The action is expanded to show any invokes and actions inside. In this example, the for-each action includes a switch action, which is also automatically expanded.



2. Collapse the action through one of the methods described above.

The action and all of its contents are collapsed.

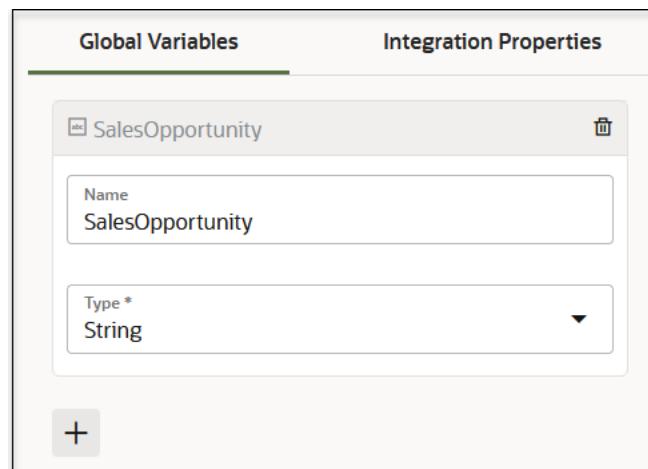


Create Global Variables

You can create complex and simple type global variables that are available for use throughout an integration (for example, when building an expression in an assign action).

User-defined complex type variables are defined based on the WSDLs/XSDs exposed in the integration by the trigger connection and any invoke connections. These variables are then available throughout the integration for assignment and usage.

1. On the right side of the integration canvas, click **Global Variables** (x).
2. Click **Add Variable** if no global variables currently exist. Otherwise, click +.
3. Enter a name and select the data type from the drop-down list. The **Object** data type is considered a complex type.



You can also use integration properties as global variables. Integration properties are user-defined properties that can be used as read-only data by actions in an integration (for example, by notifications and logs). Integration properties are external to the integration. For example, an email address used by a notification action in an integration can be externalized as a `sendTo` integration property. This enables the property to be configured without the need to edit the integration or pass the value at runtime.

4. If you add a global variable that is an **Object** data type, the panel is expanded on the left to display a **Sources** tree with all the elements from the request object.
5. Expand the tree and drag an element (root/child) to the **Type** field. This marks the current global variable type as a selected node type. The **Type** list selection changes from **Object** to the node name that you selected.

The screenshot shows the 'Input Sources' and 'Global Variables' sections of the Oracle Integration Cloud Service interface. In the 'Sources' section, 'incident *' is selected. In the 'Global Variables' section, a new variable 'SalesOpportunity' is being defined with 'Name' set to 'SalesOpportunity' and 'Type' set to 'incident'. A '+' button is visible at the bottom left of the 'Global Variables' section.

Note the following complex data type design rules:

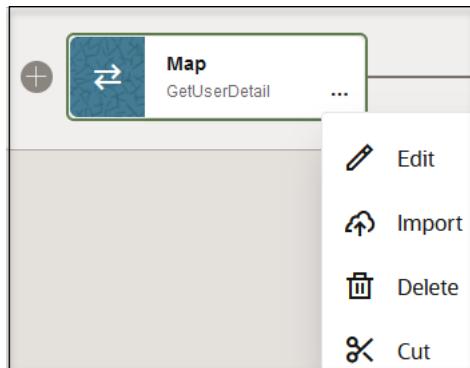
- Complex types are only based on XML root elements. If you select a nonroot element when creating a complex type global variable, a WSDL is automatically generated defining it as a root element for use throughout the integration.
 - Because complex type global variables are based on WSDLs/XSDs introduced into the integration by the trigger connection and invoke connections, there may be an impact when the trigger or invoke is deleted, modified, or regenerated. Global variables react to these design changes, which may result in invalid usage/assignments.
6. Click **+** to add more global variables. You can add a maximum of 20 global variables.
 7. Click **Save** when complete.
 8. If you want to edit a global variable, click **Delete** to delete the variable, then recreate the variable. You cannot edit global variables.

The global variables you created are now available for selection in other parts of the integration canvas, such as in an assign action.

Define Ad-Hoc Maps

You can add ad-hoc maps to an integration, when needed. You can also delete the maps that are automatically created when you add a trigger or invoke to the integration, if they are not needed.

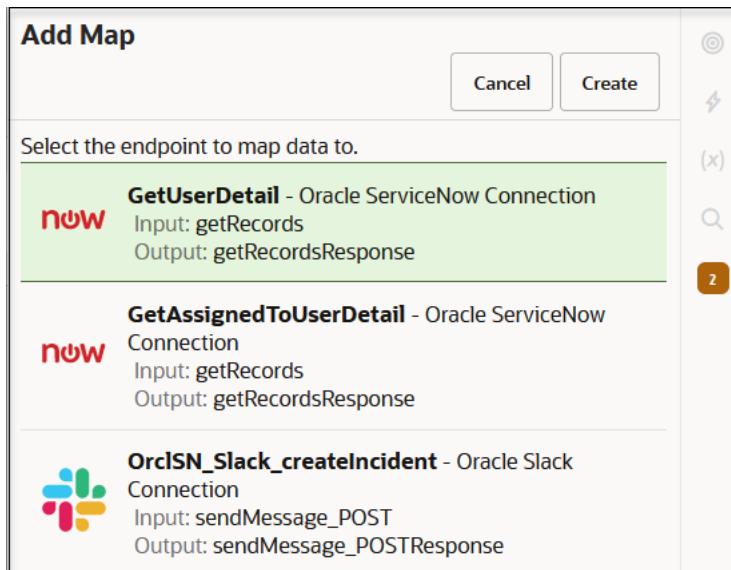
1. Delete any unneeded mappers by clicking **Actions ...**, then selecting **Delete**.



2. Add a map action to an integration in either of the following ways:

- On the side of the canvas, click **Actions** and drag the **Map** action to the appropriate location.
- Click at the location where you want to add the map action, then select **Map**.

The Add Map dialog is displayed.



3. Select the endpoint to which to map.

The mapper is displayed.

4. Map appropriate elements from the source data structure to the target data structure.
5. When complete, click **Apply** when prompted to save your changes.

Import a Map File into an Integration

You can import an XSL map file that was previously exported from the *same* integration. This action overwrites the existing mapping file.

For example, you can export the map from a specific integration, edit the XSL file per a user requirement, save it, and import it back into the same integration. You cannot import an XSL map file into an application integration that was exported from a different integration in Oracle Integration or from an application in Oracle JDeveloper.

1. Click the map action into which you want to import the file.
2. Click **Actions . . .**, then select **Import**.
3. Browse for the map file to import, then click **Import**. You only import the map file of an exported integration into Oracle Integration. You do *not* import the entire integration that includes the map file into Oracle Integration.

Add Actions to an Application Integration

You add actions as part of integration design. Actions perform the activities in an integration.

Topics:

- [Loop over Repeating Elements with a For-Each Action](#)
- [Route Expressions with Switch Action Branches](#)
- [Manage a Group of Actions and Fault Handlers with a Scope Action](#)
- [Assign Values to Scalar Variables in an Assign Action](#)
- [Loop Over Actions or Invoke Connections While a Condition is Satisfied with a While Action](#)
- [Send Notification Emails During Stages of the Integration with a Notification Action](#)
- [Build Complex Assignment Statements with a Stitch Action](#)
- [Delay Integration Processing for a Specified Time Period with a Wait Action](#)
- [Add Global Fault Handling to Integrations](#)
- [Catch Faults with a Re-throw Fault Action](#)
- [Throw Faults with a Throw New Fault Action](#)
- [Process Files in Schedule Integrations with a Stage File Action](#)
- [Receive Requests for Multiple Resources in a Single REST Adapter Trigger with a Pick Action](#)
- [Log Messages with a Logger Action](#)
- [Process Tasks in Parallel with a Parallel Action](#)
- [Invoke Oracle Cloud Infrastructure Functions Directly from an Integration with an OCI Function Action](#)
- [Publish Events in an Integration with a Publish Event Action](#)
- [Add JavaScript Functions to an Integration with a JavaScript Action](#)
- [Invoke Oracle Cloud Infrastructure Object Storage from an Integration with an OCI Object Storage Action](#)

- Add Placeholder Notes with a Note Action
- Translate an EDI Document with the B2B Action
- Convert HL7 Messages with a Healthcare Action
- Interact with Files in File Server
- Call a Robot from an Integration

Loop over Repeating Elements with a For-Each Action

The for-each action enables you to loop over a repeating element and process one or more actions within the scope of the for-each action. The number of loop iterations is based on a user-selected repeating element. For example, you may have an integration in which you download a number of files and want to loop over the output of the files. The for-each action enables you to perform this task.

Note:

When you configure a stop action inside a for-each action, the entire integration is terminated when the for-each action is processed for the first time. The for-each action is not allowed to process more than once. The stop action does not display any message describing this behavior during design time.

Creating a For-Each Action

1. Add a for-each action to an integration in either of the following ways:

- On the right side of the canvas, click **Actions**  and drag the **For Each** action to the appropriate location.
- Click  at the location where you want to add the for-each action, then select **For Each**.

The Configure For Each panel opens. This includes a **Sources** tree of elements.

2. Click  under **Configure For Each** to enter a name and optional description for the for-each action.
3. Expand the **Sources** tree to select an element.
4. Drag a repeatable element to the **Repeating Element** field. This is the element over which to loop.

 **Note:**

Note the following restrictions:

- The selected element must be repetitive. You can identify repetitive elements by the three-box icon to the left of the element name.



- Any parent of the selected element must not be repetitive.
- The data type of the selected element must be scalar.
- Global and nonglobal repeated elements can be selected.
- If you have a repeating element within another repeating element (that is, a list within a list), you must first create a for-each action and loop over the parent list. This gives you access to the child list during every iteration. You can then create a second for-each action within the scope of the first for-each action and loop over the child list.

For this example, the element over which to loop is **ICSFile**.

For every iteration of the loop, there is a single reference to the repeating element (**ICSFile**). A current element name file is required for this action to occur.

- Enter an alias for the current file of the iteration in the **Current Element Name** field.

Sources

- ▼ **schedule ***
- ⌚ startTime
- ▼ **\$getTestFile**
- ▼ **DownloadFileToICSResponse ***
- ▼ **DownloadResponse ***
- ▼ **ICSFiles ***
- ▶ **ICSFile**
- ▶ **\$getTestFile_REQUEST**
- ▶ **\$listFiles**
- ▶ **\$self**

: Configure For Each

forEachFile

Enter Description

(x) Repeating Element * ICSFile

Current Element Name * currentFile

Process items in parallel

 **Note:**

If you drag a for-each action into a schedule integration or the for-each action is not inside a while action, for-each action, scope action, and so on, an additional field called **Process items in parallel** is visible at the bottom of the dialog.

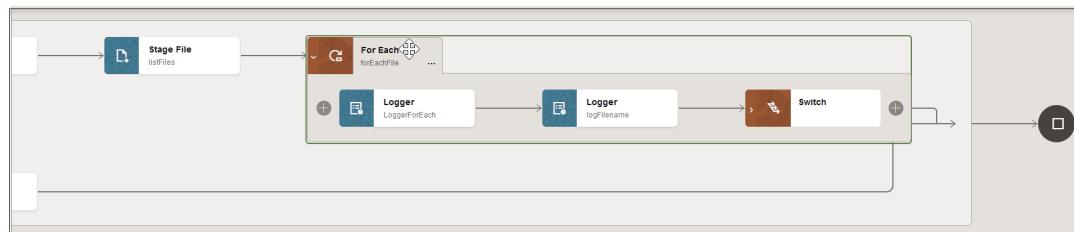
Process items in parallel

When selected, iterations of the for-each action are run in parallel. For example, if an integration processes a list of files with a for-each loop created over every file, the processing of the entire file is done in parallel for every file. If a variable is declared outside a for-each loop with **Process items in parallel** selected and updated within the for-each loop, the last updated value for that variable is available outside of the loop. Also, the degree of parallelism (DOP) is set to 1 to avoid concurrency issues.

6. Click Save.

The for-each action is displayed in the canvas. A looping arrow indicates that this action performs repetitive looping.

7. Click + inside the for-each action to add additional actions. These actions define what happens during each iteration of the loop. For this example, scope and switch actions are defined within the for-each action.



Using Global Variables in For-Each Loops

When global variables are used in for-each loops (for example, for logging purposes), they can lead to the following runtime exception when bad or invalid data is associated with the variable:

FORG0001: invalid value for cast/constructor, Cikey=13695578,
FlowId=10025204, Current Activity Key=13695578-BpAss43-BpSeq8.240-4, Current
Activity Label=assignment_ics_api_internal_variable23,

As a recommendation:

- Limit the use of global variables in for-each loops.
- Global variable handling requires database access, which results in an added performance overhead.
- Local variables can be used for logic specific to a local scope.

Tracking the Status of a For-Each Action During Runtime

You can track the status of the for-each action on the Instances page through the tracking diagram and activity stream for an activated integration.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Click the business identifier value of the integration to track.

The integration flow (including any for-each actions) is displayed. Any for-each action failures are identified by red arrows. The activity stream is displayed on the right side.

Related Documentation

The following sections provide examples of using a for-each action in an integration:

- Use Bulk Response Operations in an Integration in *Using the Salesforce Adapter with Oracle Integration 3*
- Process Large Data Sets Asynchronously with Different Bulk Import Operations in *Using the Marketo Adapter with Oracle Integration 3*
- Subscribe to Atom Feeds in a Schedule Integration in *Using the Oracle HCM Cloud Adapter with Oracle Integration 3*
- Invoke an Endpoint Dynamically in *Using the Oracle ERP Cloud Adapter with Oracle Integration 3*
- Create an Integration to Import and Process Bulk Files in *Using the FTP Adapter with Oracle Integration 3*

Route Expressions with Switch Action Branches

You can define switch action branches to add routing expressions in your integration. The switch action takes the first branch that evaluates to true. The other branches are ignored. A switch is essentially single-threaded.

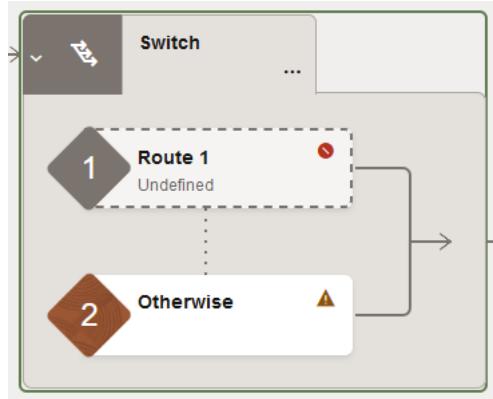
1. Add a switch action to an integration in either of the following ways:

- On the right side of the canvas, click **Actions**  and drag the **Switch** action to the appropriate location.
- Click  at the location where you want to add the switch action, then select **Switch**.

Note:

Nested switches are supported.

Two branches are automatically created:



- **Route 1** (first) branch: You must define a routing expression for this branch.
- **Otherwise** (second) branch: This branch is taken if the routing expression for the initial branch does not resolve to true.

Note:

To add more branches, click **Actions ...** in the switch action, then **Add**.

2. From the **Route 1** branch icon, double-click the branch or click **Actions ...**, then **Edit**. The Configure Route panel opens. This includes a **Sources** tree of elements. You can now define different data flows for both the defined and otherwise branches in the switch action.
3. Click under **Configure Route** to enter a name for the Route 1 branch.
4. Define a routing expression. For this example:
 - **filetype** is dragged to the upper **Value** field
 - **=** is selected as the operator
 - '**json**' is manually entered in the lower **Value** field. You can also drag a value from the **Sources** tree or select a value from the drop-down list.

You can add functions to your expression under the **Functions** tab. XPath version 2.0 functions are supported. Base 64 encode and decode functions that process data confidentially across layers and functions that return boolean results are also supported. Functions that return nonboolean values are not supported.

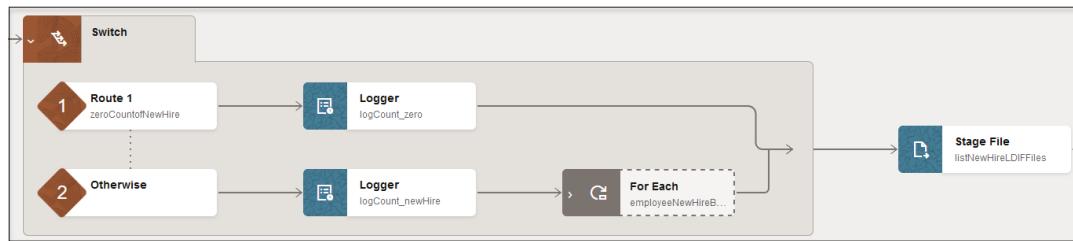
- Click **Switch to Developer Mode**  to view the complete path name.

- Click **Add condition** to add more conditions or **Add group** to add group conditions, when needed.
- Click **Save**.

For this example, the following expression is defined:

The screenshot shows the Oracle Integration Designer interface. On the left, there's a sidebar titled 'Input Sources' with a 'Sources' section containing various options like 'schedule *', '\$currentFile', 'ICSFile *', 'Properties *', 'filetype *' (which is highlighted in green), and 'directory *'. On the right, a large window titled 'Configure Route' has a 'Enter Name' field and a text input box containing 'filetype = 'json''. A magnifying glass icon is visible at the top right of the sources list.

8. Drag other actions and invokes into the branches of the switch action, when required by your integration. For this example, two switch action branches are defined.



Related Documentation

The following sections provide examples of using a switch action in an integration:

- Invoke an Endpoint Dynamically in [Using the Oracle ERP Cloud Adapter with Oracle Integration 3](#)
- Design an Asynchronous Search Operation with Pagination in [Using the Oracle NetSuite Adapter with Oracle Integration 3](#)
- Use the Extract API in a Schedule Integration in [Using the SAP Concur Adapter with Oracle Integration 3](#)
- Use the Pagination Concept in an Integration in [Using the Jira Adapter with Oracle Integration 3](#)

Manage a Group of Actions and Fault Handlers with a Scope Action

You can manage a group of actions with a scope action. The scope action is essentially a collection of child actions and invokes that can have their own fault handlers. The scope action provides the behavior context for the child elements. The elements defined in the parent scope

have local visibility inside this scope. Anything that you can do in an integration such as designing invokes, mappings, and actions can be done in a scope action.

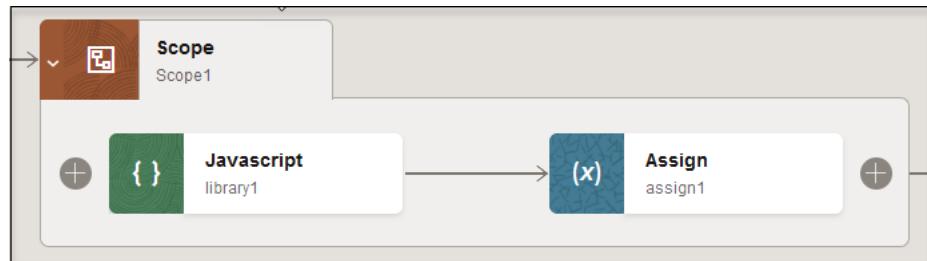
 **Note:**

Scopes can have fault handlers in which specific faults can be caught and rethrown. However, in the case of connectivity agent-based invokes, the named fault handlers are not processed. All fault handling must be done in the default fault handler.

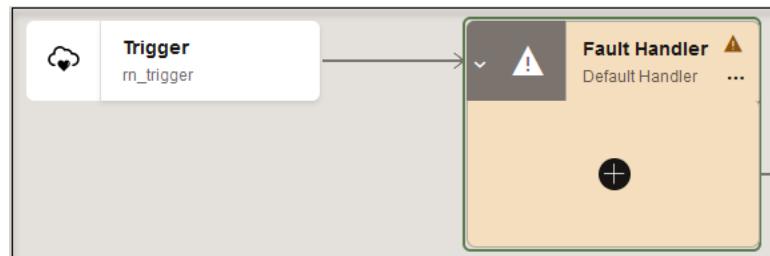
- [Creating a Scope Action](#)
- [Add Nested Scopes to a Scope Action](#)
- [Tracking the Status of a Scope Action During Runtime](#)
- [Rename a Scope](#)

Create a Scope Action

1. Add a scope action to an integration in either of the following ways:
 - On the right side of the canvas, click **Actions**  and drag the **Scope** action to the appropriate location.
 - Click  at the location where you want to add the scope action, then select **Scope**.
2. Click **Edit**  under **Edit Scope** to enter a name and optional description for the scope action.
3. Drag and design appropriate actions, mappings, and invokes into the scope action.



The scope includes a default fault handler that is accessible by clicking the scope action and selecting **Actions ...**, then **Fault Handlers**, then **Default Handler**.

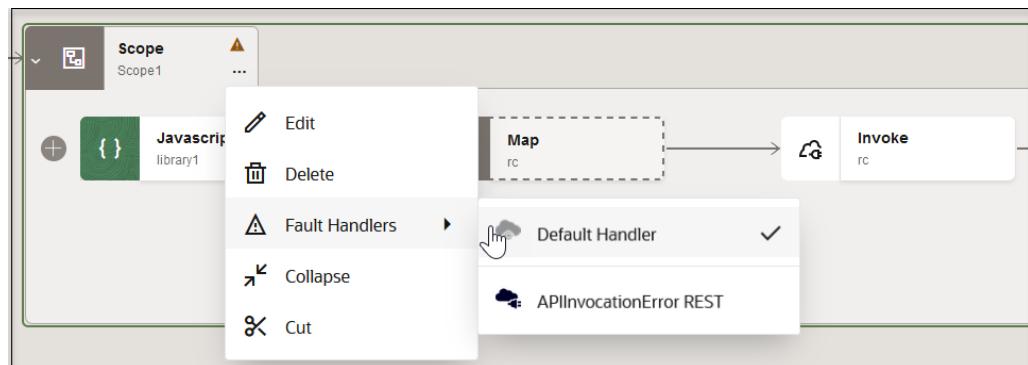


4. Click **+** to define actions on the default fault handler.

- To return to the scope action, click the fault handler, and select **Actions ...**, then **Show Scope Block 'scope_name'**.

When you add invokes to a scope, the named faults associated with the invokes are added to the default fault handler in the scope. For example:

- Add and configure an invoke connection inside the scope action (for example, a REST Adapter).
- Click the scope and select **Actions ...**, then **Fault Handlers**.
In addition to the **Default Handler**, an invoke fault handler has been added.



These are faults that the invoke can handle. The uniqueness of the named faults is defined by the qname of the fault. If there are multiple invokes that define the qname fault, the fault (at runtime) can respond to any invoke. The various invokes cannot be differentiated if their qnames are not unique.

- Select appropriate named faults.

You can also define catch all blocks for fault handling through use of the raise error action in the **Fault Handler** section of the scope action. Catch all blocks are processed if an invoke throws a fault, but there is not a specific catch named for it. See [Catch Faults with a Re-throw Fault Action](#).

- Collapse the scope action by double-clicking it. You can expand it again by double-clicking it, selecting **Actions ...**, then **Expand**, or clicking **>**.

Note:

For application integrations with multiple REST Adapter invoke connections, only one handler can be defined when there are multiple invokes in an integration with scopes. Though the fault handler options are available for each invoke in the scope, it always points to one single handler.

Add Nested Scopes to a Scope Action

You can add nested (child) scope actions to a basic scope action. This provides a more sophisticated way of organizing or separating actions into a subsection of the integration. A nested scope provides the following capabilities:

- Behaves the same way as a basic scope. It provides its own container of child actions and fault handlers.
- There is no limitation to the levels of nesting. Even a scope's fault handlers can have nested scopes.

1. Drag a scope inside an existing scope.



2. Add actions and fault handlers to the nested scope, when necessary.

Track the Status of a Scope Action During Runtime

You can track the status of the scope action in the Tracking Details page through the tracking diagram and activity stream for an activated integration. This is only possible if there is a tracking instance.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Click the business identifier value of the integration to track.

Depending upon runtime processing, several scope processing states can be displayed in the diagram:

- Scope processing succeeds and is displayed in green. Because the fault handler is not processed, the **Fault Handler** section of the scope remains hidden.
- Scope processing fails and is displayed in red, but the fault handler succeeds in handling the fault and is displayed in green. Processing continues after the scope. Because the fault handler was processed, the **Fault Handler** section of the scope is visible.
- Both the scope and fault handler fail. Both are displayed in red. Both the scope and the **Fault Handler** section are displayed.

See [Track the Status of Integration Instances](#) for additional details.

Rename a Scope

1. Open the integration that includes the scope to rename.

2. Click ↴ to collapse the scope action.
3. Click the scope and select **Actions ...**, then **Edit**.
4. Change the name and click **Save**.

Assign Values to Scalar Variables in an Assign Action

You can assign values to scalar variables in integrations with an assign action.

 **Note:**

- Variables created inside a scope action or a looping action (for example, a for-each or while action) are not directly accessible outside the scope/looping action. To access the variables (local) outside a scope/looping action, create a global variable using an assign action above the scope/looping action. Assign the local variable to this global variable and then use it outside the scope/loop action.
- Values cannot be assigned to other variable types, such as complex types.

1. Add an assign action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Assign** action to the appropriate location.
- Click  at the location where you want to add the assign action, then select **Assign**. The Configure Assign panel opens.

2. Click **Edit**  under **Configure Assign** to enter a name and optional description for the assign action.

3. Click the  sign, then select a data type:

- **String:** Create a string data type.
- **Fault:** Catch and aggregate faults in parallel processing actions in schedule integrations. This operation is only supported in schedule integrations.

4. If you select **String**:

- Enter a string variable name or select an existing variable from the list.

The screenshot shows the Oracle Integration Workbench interface. On the left, there's a sidebar with 'Input Sources' and 'Functions'. Below that is a list of 'Sources' including 'schedule *', 'startTime', '\$self', '\$file2get', '\$tracking_var_1', '\$tracking_var_2', and '\$tracking_var_3'. On the right, the main area is titled 'Configure Assignment' with a sub-section 'assignTestVars'. It has a text input field 'Enter Description' and a note: 'Specify as many variables for this assignment as you need, and configure their values.' Below this is a configuration panel for variable 'l1_1'. It shows 'Variable l1_1' and an assignment '= Variables' with 'ftpReadFolder' selected. A dropdown menu shows other variables: downloadDir, extensionType, file2get, ftpReadFolder (selected), and isZipfile.

The **Operations** field defaults to an equal sign (=) and cannot be changed.

- Enter a value between the single quotes.

This screenshot shows a configuration dialog for an 'Assign' operation. It has three fields: 'Variable ftpReadFolder', 'Operation Assign', and 'Value '/akt_ftp/read''. The 'Value' field has a red underline under 'akt_ftp'.

- Add more string variables, when needed.

Configure Assignment

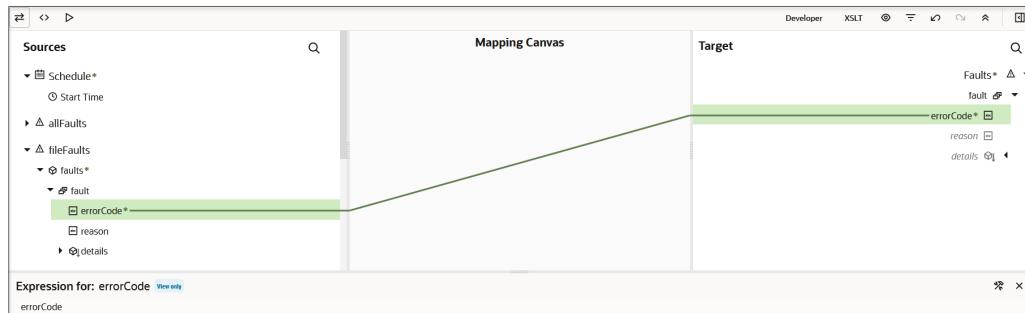
`assignTestVars`

Enter Description

Specify as many variables for this assignment as you need, and configure their values.

<input type="checkbox"/>	<code>downloadDir = '/stage_read/'</code>
<input type="checkbox"/>	<code>isZipfile = ends-with(file2get, 'zip')</code>
<input type="checkbox"/>	<code>testFile = file2get</code>
<input type="checkbox"/>	<code>ftpReadFolder = '/akt_ftp/read/'</code>
<input type="checkbox"/>	<code>extensionType = substring-after(file2get, ':')</code>

5. If you select **Fault**:
 - a. Enter a fault variable name or select an existing fault from the list.
 - b. Select an operation from the **Operation** list.
 - **Append**: Adds a new fault to the end of the fault list.
 - **Clear**: Removes all entries in the fault list.
 - c. If you selected **Append**, click **Edit**  to select a fault from the mapper in the **Value** column. This option is not available with the **Clear** operation.



6. Add more fault variables, as needed.
7. Click **Save**.

Variable assignments can be of great complexity. You can use assignments in switch actions and maps. For example, the upper branch of a switch action is taken (if `$FetchContactAssign = "failed"`), which processes a **contactCreateProcessing** scope. Otherwise, a **contactUpdateSkip** scope is processed.

You can also configure the primary tracking variable and both custom field tracking variables (update and access values). You can map tracking variables to output variables or create complex expressions for an assign or switch activity.

- All tracking variables are of type string (all that assignments support).
- All three tracking variable entries are present even if you choose not to model them. The name and XPath can be empty for tracking.
- Editing or deleting the tracking variables only updates the name and XPath nodes in that particular tracking variable element.
- The tracking variables have static names. Therefore, it is possible to set a tracking variable somewhere in the flow, but not initialize it with a value and a name in the Tracking dialog.
- The primary tracking variable cannot be assigned any value in between the flow.
- You cannot create a new variable with the same names as any of the statically named tracking variables.

Related Documentation

The following sections provide examples of using an assign action in an integration:

- Retrieve the Name and IDs from Account Records with the queryMore Operation in *Using the Salesforce Adapter with Oracle Integration 3*
- Design a Basic Asynchronous CRUD Operation in *Using the Oracle NetSuite Adapter with Oracle Integration 3*
- Use the Extract API in a Schedule Integration in *Using the SAP Concur Adapter with Oracle Integration 3*
- Use the Pagination Concept in an Integration in *Using the Jira Adapter with Oracle Integration 3*

Loop Over Actions or Invoke Connections While a Condition is Satisfied with a While Action

The while action enables you to loop over actions or invoke connections as long as a specific condition is met. You define the condition for the while loop in the Expression Builder. The while action is available in both scheduled and application integrations.

Creating a While Action

Note:

Variables used in while action statements can be of several types, including number, string, and boolean types.

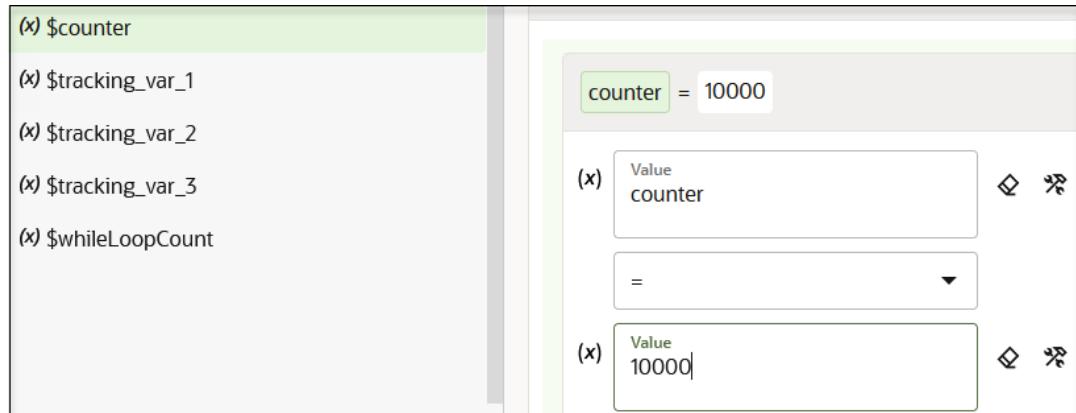
1. Add a while action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **While** action to the appropriate location.
- Click  at the location where you want to add the while action, then select **While**.

The Configure While panel opens. This includes a **Sources** tree of elements.

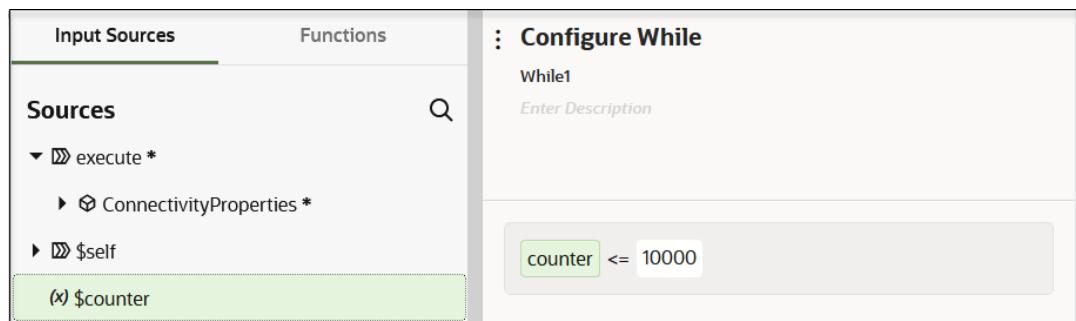
2. Click  under **Configure While** to enter a name and optional description for the while action.

3. Manually enter, select from the drop-down list, or drag an element from the **Sources** tree to the **Value** field. You can also add functions.
4. Select an operator.
5. Enter a value in the **Value** field.

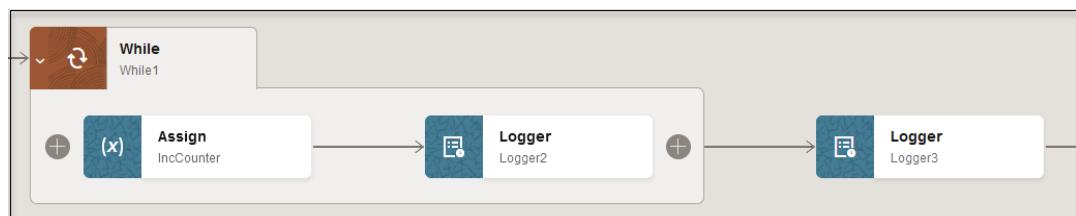


6. Click **Save**.

For this example, while the counter is less than or equal to 10000, the integration loops over the condition. When the condition is not met, the looping ends.



7. Drag invoke connections or other actions for configuration to the + sign that is displayed inside the while action. These invoke connections and actions are processed as long as the condition set in the while action is met.



8. If you want to edit the name or expression, double-click the **While** icon or select **Actions . . .**, then **Edit**.

 **Note:**

Deleting a while action has no impact on downstream processing of the integration because the while action does not have any output. Any changes in the upstream actions in the integration that impact the while condition result in the display of a warning icon on the while action.

Track the Status of a While Action During Runtime

During runtime, you can track the status of the while action in the Instances page through the tracking diagram and activity stream for an activated integration.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Click the business identifier value of the integration to track.

The integration flow (including any while actions) is displayed. Any while action failures are identified by red.

Related Documentation

The following sections provide examples of using a while action in an integration:

- Retrieve the Name and IDs from Account Records with the queryMore Operation in *Using the Salesforce Adapter with Oracle Integration 3*
- Retrieve Issue Details Using the Jira Query Language (JQL) in *Using the Jira Adapter with Oracle Integration 3*
- Use Pagination in an Integration in *Using the Shopify Adapter with Oracle Integration 3*

Send Notification Emails During Stages of the Integration with a Notification Action

You can send a notification email to relevant users at specific points in the processing of an integration. You can set the to, from, and subject parts of an email. You can create the body part of an email using parameters defined in the Expression Builder. You can also add attachments to the email if your integration includes them. The total size limit on a notification email is 1 MB for Oracle Integration and 2 MB for Oracle Integration Generation 2. Both the email body and attachment are considered in calculating the total size.

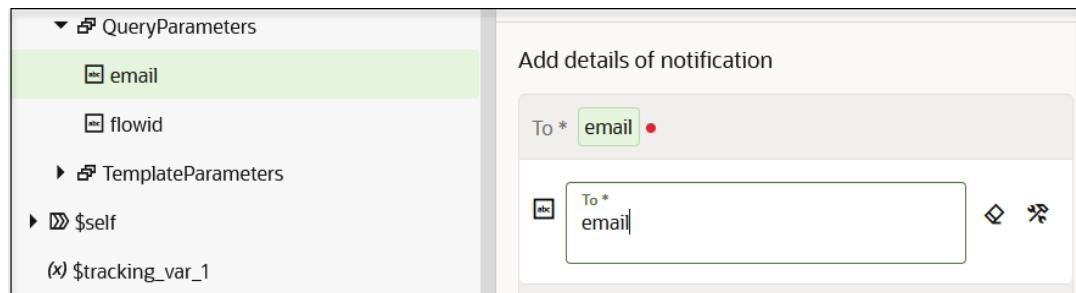
1. Add a notification action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Notification** action to the appropriate location.
- Click  at the location where you want to add the notification action, then select **Notification**.

2. Click **Edit**  under **Configure Notification** to enter a name and optional description for the notification action.
3. Click the **To** field.

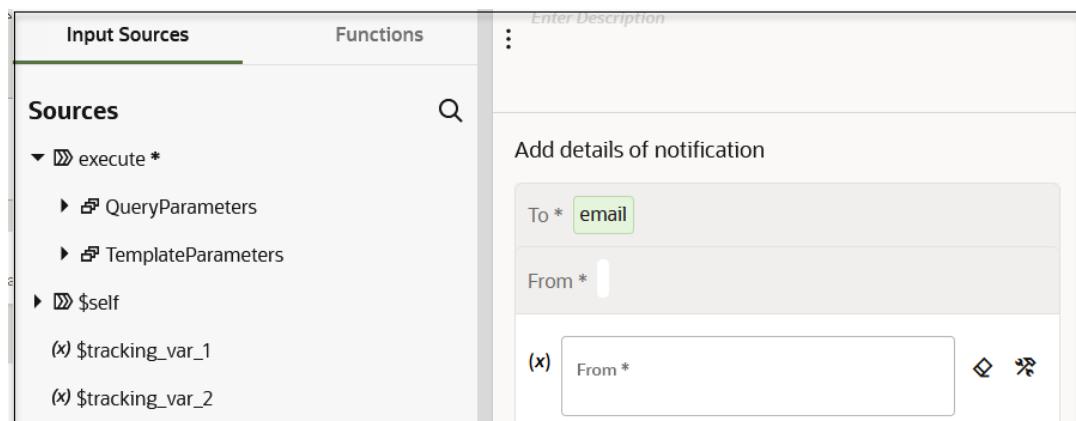
This opens the **Sources** tree for defining an expression and a second **To** field for editing immediately after the first field.

4. Manually enter an email address, select from the drop-down list, or drag an element from the **Sources** tree to the **To** field:



5. Click the **From** field.

This opens a second **From** field for editing immediately after the first field.



6. Manually enter an email address, select from the drop-down list, or drag an element from the **Sources** tree. For this example, an **email** element is dragged from the **Sources** tree.

You can provide an email address in the **From** field that is approved as the sender for service failure alerts, system status reports, and integration error reports. You configure the approval email address to use in the **From** field on the Notifications page that is accessible by selecting **Settings**, and then **Notifications**.

Note:

If you send emails with the notification action and the **From** address is not one of the approved senders configured on the Notifications page, the **From** address defaults to the approved sender address of no-reply@mail.integration.region.ocp.oraclecloud.com. For example, if your region is Ashburn, the address is no-reply@mail.integration.us-ashburn-1.ocp.oraclecloud.com. See [Configure Notification Emails](#).

See [Send System Status Reports with Notification Emails](#).

7. Click the **Subject** field.

This opens a second **Subject** field for editing immediately after the first field.

8. Manually enter a subject in plain text, select from the drop-down list, or build an expression from the **Source** or **Functions** tree. To drag a function, ensure that you first select **Switch to Developer View** .

For this example, a **concat** function is selected for use from the **Functions** section.

```
concat ( 'Hello', name )
```

9. Click the **Body** field.

This opens a second **Body** field for editing immediately after the first field.

10. Enter a message using plain text and plain HTML formatting that you create in a separate HTML editing tool and paste into this field or parameters that you create in the table immediately after this field. After creating parameters, enter them inside { } brackets.

```
<html>
<body>
<p>Hello <strong><span style="color: #2925BF;">{name}</span></strong></p>
<p><span style="color: #298022;"><strong>Welcome to Oracle Integration!</strong></span></p>
</body>
</html>
```

11. Click **+** to add parameters.
12. Enter a parameter name and description or build an expression.
13. Click **+** to add attachments.
14. Select the attachments to add.
15. In the **Attachments** section, click **+** to open a page to select an attachment file.

You can edit or delete attachments once added. Examples of attachments may include file reference attachments that are available for selection:

- An attachment from a REST Adapter connection
- A file reference from a stage file write operation
- A file reference from an FTP Adapter download operation

16. When complete, the Configure Notification panel looks as follows for this example:

Configure Notification

sendEmail

Enter Description

Add details of notification

To * email

From * email

Subject * concat ('Hello' , name)

Body *

```
<html> <body> <p>Hello <strong><span style="color: #2925BF;">{name},</span></strong></p> <p><span style="color: #298022;"><strong>Welcome to Oracle Integration Cloud!</strong></span></p></body> </html>
```

Parameters

name name

+

▼ Attachments

+

17. Click **Save**.

When the email notification is received during integration runtime, the parameter `name` is replaced with a dynamic value.

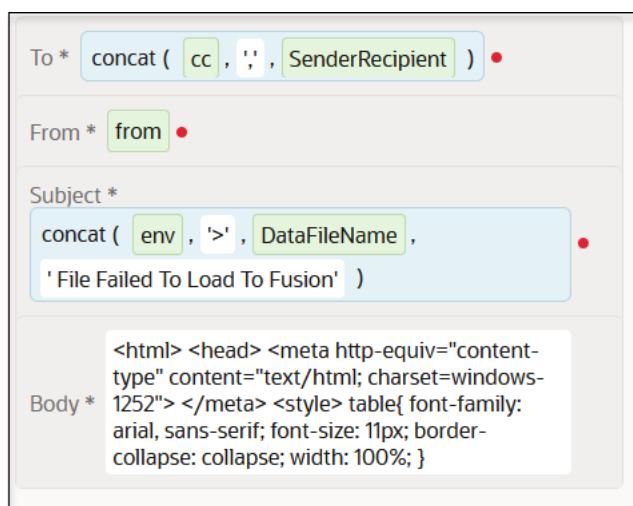
 **Note:**

- Deleting the notification action does not impact downstream activities because a notification does not have any output. Changes in the upstream activities impact the notification when they are used either in the **From**, **To**, or **Subject** fields or in the body parameters. For example, if the `name` example used in this section is modified, the parameter assignment become invalid.
- Notification actions are treated as asynchronous actions with no failure. For example, assume you include a notification action in an integration and disable the sendmail service on your host, which prevents you from receiving an email notification. The integration instance appears as completed on the Instances page and there is no error message in the instance. This is the expected behavior. You can only see an issue with the instance if you open the integration instance and view the notification action.

Set the content-type in a Notification Action

You can set the `content-type` in an email body of a notification action.

Code the `content-type` as follows:



Mapper and Assign Action Capabilities	Stitch Action Capabilities
Mapper: Only generates full message payloads. If you attempt to map into an existing message payload, a full replacement of that payload occurs.	Supports both partial and full replacement of the message payload.
Assign action: Limited to scalar type variables. Complex objects or full payloads are not supported.	Supports both scalar and complex type variables.

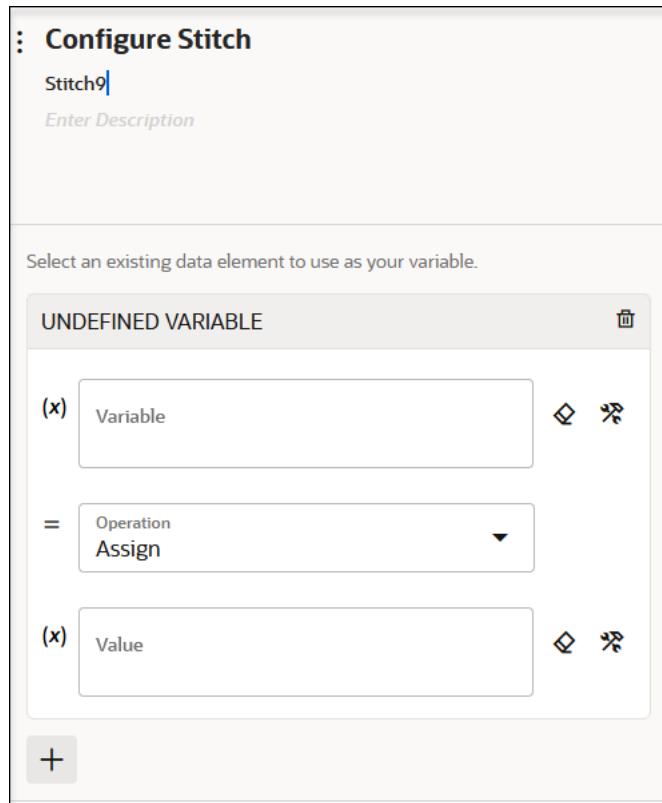
You can perform the following complex assignments (operations) on variables (and child elements of variables) with the stitch action:

- Append: Appends at the end of the repeating/unbounded target element, the selected element, or the value. For example, you have an existing purchase order payload containing five lines and want to add a sixth line. The stitch action enables you to append a sixth line to the existing array of lines in the purchase order.
- Assign: Places the selected value/element/attribute into the target element/attribute, overriding any existing data in the target element/attribute. For example, you want to change the current address in an existing purchase order. The stitch action enables you to change the address. You can either map fields individually or copy the address object itself.
- Remove: Removes the target element/attribute from the variable. For example, you have an existing purchase order payload and want to remove the price to enable the end point application to calculate a new price. The stitch action enables you to remove the price. For repeating/unbounded elements, all instances are removed unless a specific instance is selected by index or predicates.

1. Add a stitch action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Data Stitch** action to the appropriate location.
- Click  at the location where you want to add the data stitch action, then select **Data Stitch**.

The Configure Stitch panel opens.

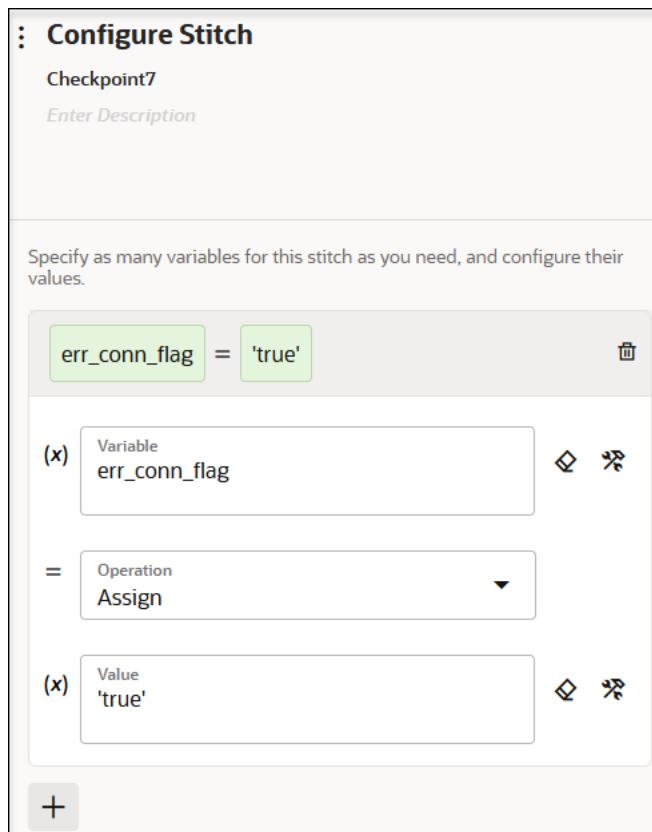


2. Click **Edit**  under **Configure Stitch** to enter a name and optional description for the stitch action.
3. Click the **Variable** field and manually enter a variable name, select a variable from the drop-down list, or click **Switch to Developer View**  to open the **Sources** tree for dragging a variable name.

The screenshot shows the Oracle Integration Designer's 'Configure Stitch' interface. On the left, there's a sidebar with 'Input Sources' and 'Functions' tabs, and a 'Sources' tree view containing various integration statements like '\$GetFulfillmentRequest_OM_Trigger' and '\$UpdateAssetStatus_IB_Invoke'. The main panel is titled 'Configure Stitch' for 'Stitch9'. It includes a search bar and a note to 'Select an existing data element to use as your variable.' Below this is a 'UNDEFINED VARIABLE' section with fields for 'Variable' (containing '(x)'), 'Operation' (set to 'Assign'), and 'Value'. A '+' button is at the bottom right.

Global variables and business identifier tracking variables are displayed for selection. A data element is a subcomponent of a variable. A complex variable conforms to a series of data elements.

4. From the **Operation** list, select the operation to perform on the variable:
 - **Append**
 - **Assign**
 - **Remove**
5. Click the **Value** field and specify a value in any of the following ways:
 - Entering manually.
 - Selecting from the list.
 - Clicking **Switch to Developer View** to open the **Sources** tree.
- If you selected the **Remove** operation, the **Value** field is not displayed.
6. Click **+** if you want to create additional complex assignment statements.
7. When complete, click **Save** to save your statements and close the Configure Stitch panel. For this example, the following statement was created.



You can also define the sequence of variables to update. For example, if you want to copy an address, and then override a child element such as street, place these statements in the correct order of processing.

Delay Integration Processing for a Specified Time Period with a Wait Action

The wait action enables you to delay the processing of an integration for a specified period of time. Use this action in schedule integrations, asynchronous integrations, synchronous integrations, and fire-and-forget integrations. A typical use for this action is to invoke a specific operation at a certain time. You can specify literal values or build XPath expressions that you can configure with property values during runtime.

Create a Wait Action

- Add a wait action to an integration in either of the following ways:

- On the side of the canvas, click **Actions** and drag the **Wait** action to the appropriate location.
- Click at the location where you want to add the wait action, then select **Wait**.

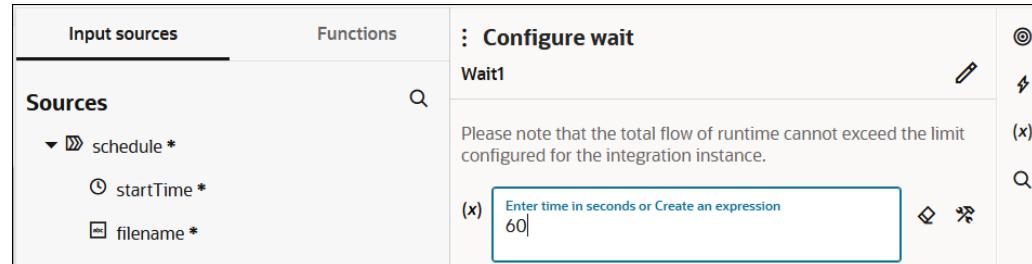
The Configure wait panel opens.

- Click **Edit** under **Configure wait** to enter a name and optional description for the wait action.

Configure the time to wait before processing the integration in either of two ways:

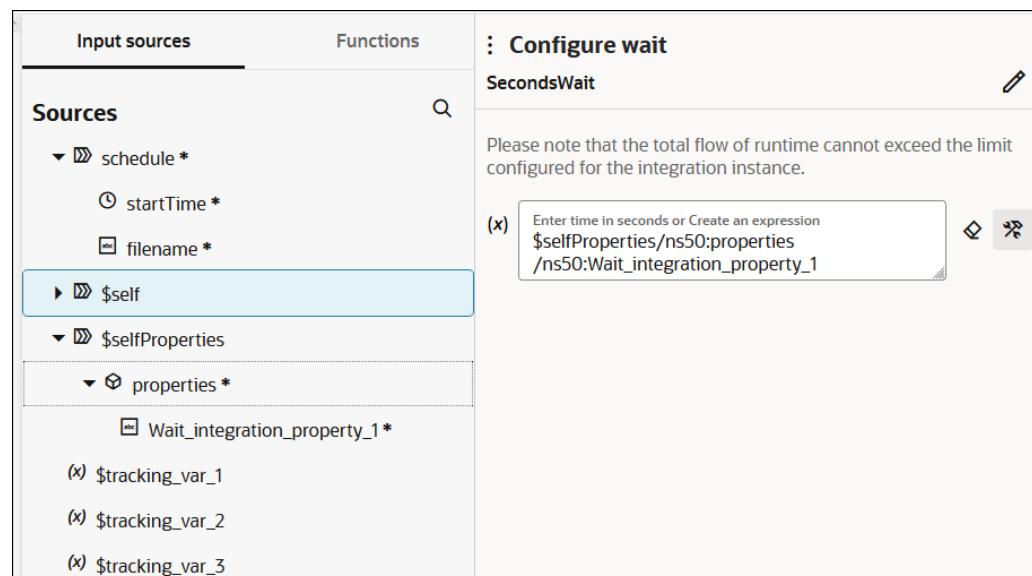
- If you want to use literal (hard coded) values:

- Enter the number of seconds to wait before processing the integration. Literal values can only be in seconds, with no upper limit. The wait time cannot exceed the total running time of the integration.

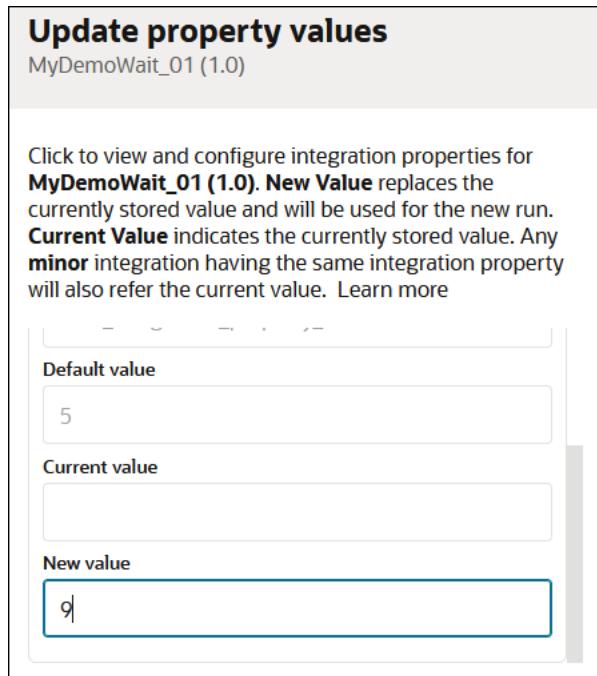


- If you want to enter an XPath expression:
This option enables you to update a wait action parameter with an integration property value after the integration is activated. This update does not require a re-activation of the integration.

- Drag an element from the **Input sources** section.



- Complete design of your integration and exit the integration canvas.
- Activate the integration.
- Hover your cursor over the integration, and select **Actions ...**, then select **Update property values**.
- Enter a value to use at runtime in the **New value** field (for this example, 9 is entered). This value overrides the value provided in the **Default value** field.



- f. Run the integration and view the activity stream.
- g. Click **View** ⓘ for the action in front of the wait action (for this example, a logger action) and note that the integration property value is **9** and the wait action was initiated at **02:36:32** and the next action (for this example, a second logger action) was initiated at **02:36:41** (nine seconds later).

Activity stream
Instance Id: bZNOLsQ-Ee2SBdNGAD0f8g

Tracing level: **Production**

- Fri Mar 17 02:36:32.461 AM IST 2023
Triggered a Scheduled Run
- 02:36:32.533 AM, 51ms
Scheduled Run started executing
- 02:36:32.584 AM
Message started processing
- 02:36:32.661 AM, 1ms
Logger Logger1
- Before Wait : Wait Integration Property Value is : 9
- 02:36:32.662 AM, 9s 2ms
Wait Wait1 triggered
- 02:36:41.664 AM
Logger Logger2
- 02:36:41.664 AM
Processing completed successfully

Track the Status of a Wait Action During Runtime

During runtime, you can track the status of the wait action on the Tracking page through the tracking diagram and activity stream for an activated integration.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Click the business identifier value of the integration to track.
The integration flow (including any wait actions) is displayed.

Related Documentation

The following sections provide examples of using a wait action in an integration:

- Use the Extract API in a Schedule Integration in *Using the SAP Concur Adapter with Oracle Integration 3*
- Process RaaS Reports that Support the Delivery of Data in CSV Format in *Using the Workday Adapter with Oracle Integration 3*
- Export Candidate Data from Oracle Talent Acquisition Cloud (Taleo EE) in *Using the Oracle Taleo Enterprise Edition Adapter with Oracle Integration 3*

Add Global Fault Handling to Integrations

You can add global fault handling to integrations. This functionality enables you to direct business faults back to the caller or apply business logic before sending faults to the error

handling framework. You can add fault handling to any integration type (for example, asynchronous, synchronous, or scheduled fire-and-forget (no response expected)).

Add a Global Fault

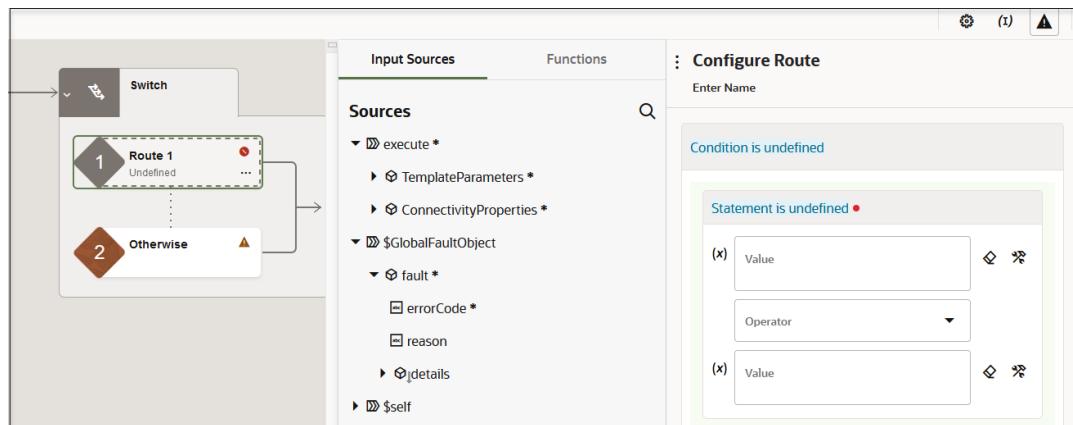
1. Design an integration. The integration does not need to be complete. You can add fault handling at any time. However, the integration must include an invoke connection.

2. In the integration canvas, click **View Global Fault Handler** .

The Global Fault Handler page is displayed. The initial trigger in your integration is automatically connected to an initial **Re-throw Fault** action. However, you can add and delete other **Re-throw Fault** actions. The **Re-throw Fault** action does not respond back to the trigger. Instead, details collected by the **Re-throw Fault** action are sent to the error handling framework.



3. Click **Actions**  to add actions to design specific fault handling logic in the integration. For example, you can add **Switch**, **Stop**, or additional actions, when needed.
4. To build an expression to capture fault handling information, select the action, then click **...**, then **Edit**.



The screenshot shows the Oracle Integration Designer interface. On the left, there is a route diagram with a 'Switch' action. One of its branches leads to a 'Route 1' action, which has a red exclamation mark indicating an error. Another branch leads to an 'Otherwise' action. On the right, a configuration dialog for the 'Configure Route' is open. It shows a 'Sources' section with various options like 'execute *', 'fault *', and '\$self'. Below it, a 'Condition' section is visible with the message 'Condition is undefined'. A 'Statement' section contains two input fields: '(x) Value' and '(x) Value', each with a dropdown menu labeled 'Operator'.

Note:

The Expression Builder includes the following functions under **Functions > Integration Cloud** for designing fault handling:

- `getFaultAsString` (returns the fault as a string value)
- `getFaultAsXML` (the fault as an XML element)
- `getFaultName` (returns the fault name)
- `getFaultedActionName` (returns the fault of the action)
- `getFlowId` (returns the flow ID of the integration)
- `lookupValue`

These functions are only available within the Expression Builder in integrations.

5. Build an expression to capture fault handling information. For example:

```
$TargetApplicationObject1/nssrcmpr:fault/nssrcmpr:details = "ERROR"
```

6. Drag a **Stop** action to the **Otherwise** branch of the switch action.

The fault handling logic is now complete. For this example, if an error occurs, error details are captured and sent to the error handling framework (as indicated by the **Re-throw Fault** action). If no error occurs, the fault handling stops and nothing is sent to the error handling framework (as indicated by the **Stop** action).

7. Click **Save**.

8. Click **View Main Integration**  to return to edit mode in the integration canvas.
9. Complete the design of the integration.
10. Activate the integration.

Track the Status of a Global Fault During Runtime

During runtime, you can track the status of global faults on the Instances page through the tracking diagram and activity stream for an activated integration. This is only possible if there is a tracking instance.

1. In the navigation pane, click **Observability**, then **Instances**.
2. On the Instances page, click the business identifier value of the integration to track.

Global fault handling is only invoked if there is a failure in the integration flow. Failures are identified by red.

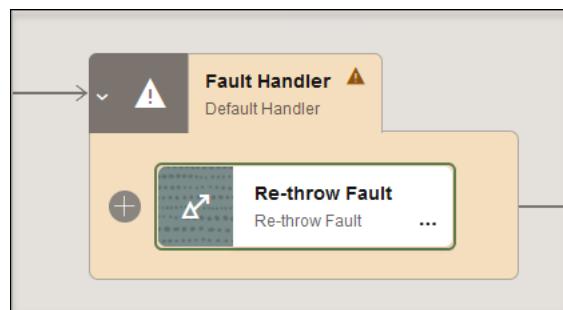
If the global fault handler successfully handled the error, the integration is displayed as **Succeeded** on the Instances page.

Catch Faults with a Re-throw Fault Action

You can send failed messages to the error hospital for further analysis with a re-throw fault action. If the integration contains a defined global fault, the error captured by the re-throw fault action is sent through the global fault and onto the error hospital for analysis. If no global fault is defined, the fault is sent directly to the error hospital for analysis. The re-throw fault action can only be placed inside the fault handler section of a scope action. The re-throw fault action

operates as a catch all block and is processed if a fault is thrown by an invoke action in the scope. However, the re-throw fault action does not have a specific catch named for it. The following example describes how to define a re-throw fault action in a scope action.

1. Create an integration that includes a scope action. See [Manage a Group of Actions and Fault Handlers with a Scope Action](#).
2. Double-click the scope.
3. Select **Actions . . .**, then **Fault Handlers**, and then **Default Handler**.
The scope action is now in fault design mode.
4. Add a re-throw fault action to an integration in either of the following ways:
 - On the right side of the canvas, click **Actions**  and drag the **Re-throw Fault** action to the + sign.
 - Click  at the location where you want to add the re-throw fault action, then select **Re-throw Fault**.
This creates a re-throw fault action in the fault handling section of the scope action.



Any faults captured by this action are passed to the error hospital for analysis. Because of this fault, the integration flow is terminated.

5. Double-click the scope action to return to design mode.
The **Re-throw Fault** action is hidden.
6. Design your scope action.
7. To return to fault mode, double-click the scope.
8. Select **Actions . . .**, then **Fault Handlers**, and then **Default Handler**.

Throw Faults with a Throw New Fault Action

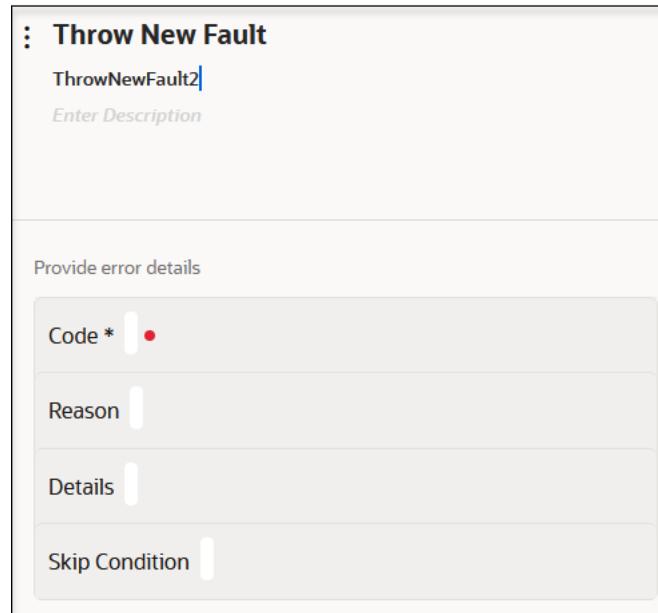
You can create and throw your own faults in an integration with a throw new fault action. During configuration of this action, you define the condition under which to throw the fault and the point in the integration at which to throw the fault. You can add this action at the end of a block (for example, a for-each action, switch action, and so on). Nothing can be dropped after this action in the block.

1. Add a throw new fault action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Throw New Fault** action to the appropriate location.

- Click  at the location where you want to add the throw new fault action, then select **Throw New Fault**.
2. Click **Edit**  under **Throw New Fault** to enter a name and optional description for the throw new fault action.

The Throw New Fault panel is displayed.



The screenshot shows the 'Throw New Fault' configuration dialog. At the top, there's a field labeled 'Name' containing 'ThrowNewFault2'. Below it is a 'Description' field with placeholder text 'Enter Description'. The main section is titled 'Provide error details' and contains four fields: 'Code *' (with a red asterisk), 'Reason', 'Details', and 'Skip Condition'. Each of these fields has a small input field with a vertical line and a red dot indicating it's required.

3. Click a field to open the **Sources** panel and specify values in any of the following ways:
- Enter manually.
 - Select from the list.
 - Click **Switch to Developer View**  to drag a value from the **Sources** panel.

4. Complete the fields:

Element	Description
Code	Create a code value. This is a mandatory field.
Reason	Define a reason for the error.
Details	Define additional error details.
Skip Condition	Define a condition to prevent the fault from being thrown.

5. Click **Save**

If a skip condition is not defined, the throw new fault action is displayed with a dashed line connecting to the next action. This signifies that the action continues processing only after processing the fault.



If a skip condition is defined, the line connecting to the next action is solid. The solid line indicates that it is possible that the processing of the integration bypasses the fault and goes straight through to the next action.

Deleting the throw new fault action has no impact on downstream activities because this error does not have any output. Any changes to upstream activities triggers a throw new fault action validation because both the **Code**, **Reason**, and **Details** fields in the Throw New Fault page can point to flow input or upstream outputs.

Process Files in Schedule Integrations with a Stage File Action

You can use the stage file action to process files or file references in schedule integrations. The stage file action can process each file downloaded by the FTP Adapter. The stage file action can read (and remove any trailer), write, zip, unzip, encrypt, decrypt, and list files in a staged location known to Oracle Integration.

The stage file action can also read (and remove any trailer) and unzip referenced files in a staged location. The stage file action is similar in functionality to adapters. However, unlike adapters, you do not need to create a connection to use the stage file action. The stage file action has no credential properties and security policies. The stage file action also differs from the File Adapter and FTP Adapter in that it provides the ability to define a file format for read and write operations. For the stage file action to process or act upon files and attachments, they must be available in Oracle Integration. Make the files available in Oracle Integration either using the download operation in the FTP Adapter or consuming the SOAP/REST APIs that return multipart or MTOM attachments. Oracle Integration flows exposed as REST endpoints using the REST Adapter and exposing an interface to accept multipart attachments also automatically stage files in Oracle Integration when the requests are posted to the endpoint.

For example, you may include a stage file action in an integration as follows:

- Configure an FTP Adapter with the following settings:
 - **Download File** operation
 - **Unzip the File** option
 - Input directory and download directory path
- Because the ZIP file may contain multiple files, configure a for-each action after the FTP Adapter to iterate over repeated elements.
- To read each file from the input directory, configure a stage file action after the for-each action to read each file from the input directory as follows:
 - In the Expression Builder, specify the file name and directory from which to read the file.

- Specify the schema file by loading a comma-separated value (CSV) file that contains the data structure.

 **Note:**

- Do not use special characters in schema names (such as #). Otherwise, integration activation fails.
- The stage file action only supports the .zip file format. For example, if the input file is .gz format, Oracle Integration cannot unzip or read the contents of the file.

Process File References

You can specify a file reference to process when you select to read the entire file, read a file in segments, or unzip a file.

This feature provides the following benefits:

- Processes upstream operations that provide a file reference. For example, a REST Adapter connection enables you to download an attachment into an attachment folder. The REST Adapter provides a file reference, but does not provide a file name or directory. The stage file action provides both these options.

The following operations provide file references:

- Attachment reference (REST Adapter attachments)
- Stream reference (REST Adapter invoke response)
- MTOM (for a SOAP Adapter invoke connection response)
- FileReference (for an FTP Adapter)
- Provides a prerequisite to process encrypted content. This means that other adapters do not need to duplicate the decryption operation.
- Encrypts content as a post-processing operation. Therefore, other adapters do not need to duplicate the encryption operation.

Create Local Files

Note the following details about local file behavior:

- Local Oracle Integration folders can only be created using a stage file write operation from within the integration.
- Within the integration scope, the file is available for further processing.
- Using stage file operations such as read, write, and others enables you to read the contents in the scope in which the file is available.
- The file is not visible outside the scope in which it was created.
- You can use stage file write-related variables in mapping operations to point to this virtual file.
- Once the integration moves outside the scope of file visibility, the local file is deleted.

Configure a Stage File Action

You can configure a stage file action in the Stage File panel.

Restrictions on Using Stage File Action Operations with the File/Attachment Features of the Connectivity Agent

When configuring the stage file action in the Stage File panel, note that there are restrictions on using some operations with the connectivity agent.

- List File operation: Files uploaded through the connectivity agent are not available with the List File operation.
- Read Entire File and Read File in Segments operations: Files uploaded through the connectivity agent can only be read with a file reference.
- Zip File operation: Files uploaded through the connectivity agent are not available with the Zip File operation.
- Unzip File operation: Files uploaded through the connectivity agent can only be unzipped using a file reference.

Open the Stage File Panel

Note:

Stage file operations cannot share a schema root element that defines the record structure if they are working with different file formats and record structures. Either the target namespace or the root element name must be unique while parsing different file formats and records structures.

1. Add a stage file action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Stage File** action to the appropriate location.
- Click  at the location where you want to add the stage file action, then select **Stage File**.

2. Complete the fields on the Basic Info page, and click **>**.

The Configure Operation page is displayed.

- [Configure Stage File Operations](#)
- [Define the Schema File](#)
- [Define the Schema Format](#)
- [Review the Stage File Action in the Integration Canvas](#)

Configure Stage File Operations

The Configure Operation page enables you to define the file operations to perform. You can select to read an entire file, read files in chunked sizes, unzip a file, write a file, zip a file, or list a file. You use the Expression Builder to build the specific details of your operation. The Expression Builder shows all upstream sources (including assignments, for-each actions, invoke connections, and so on) for you to specify these details. You can also select to encrypt

and decrypt files with a Pretty Good Privacy (PGP) authentication key that you configured on the Upload Certificate page. See [Upload a Certificate to Connect with External Services](#).

1. From the drop-down list, select the stage file operation to perform. The page refreshes to display fields appropriate to your selection. There are restrictions when using the connectivity agent with some stage file action operations. See [Restrictions on Using Stage File Action Operations with the File/Attachment Features of the Connectivity Agent](#).
 - [Read Entire File](#)
 - [Read File in Segments](#) (for chunking files)
 - [Write File](#)
 - [Zip File](#)
 - [Unzip File](#)
 - [Encrypt File](#)
 - [Decrypt File](#)
 - [List File](#)

Read Entire File

Table 5-1 - Read Entire File

Property	Description
Configure File Reference	<ul style="list-style-type: none"> • Yes: Select to process an upstream operation that provides a file reference. Once selected, you specify the file reference. • No: Select to process a file name.
Specify the File Reference (Appears if you select Yes for the Configure File Reference .)	<p>Click the field and specify a file reference in any of the following ways:</p> <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. <p>• Clicking Switch to Developer View  to open the Sources tree.</p>
Specify the File Name (Appears if you select No for the Configure File Reference .)	<p>Click the field and specify a file name (for example, /compress:schedule/compress:start Time) in any of the following ways:</p> <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. <p>• Clicking Switch to Developer View  to open the Sources tree.</p> <p>Note: The file size must be less than 10 MB. For files greater than 10 MB, use the Read File in Segments operation.</p>

Table 5-1 (Cont.) - Read Entire File

Property	Description
Specify the Directory to read from	Click the field and specify a directory from which to read files (for example, / compress:schedule/compress:start Time) in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. • Clicking Switch to Developer View  to open the Sources tree.
Remove Trailer	Select to not remove the trailer, to remove the last row, or to remove the last <i>n</i> rows.
Decrypt	Click the check box, then select the private key of the target location to use to decrypt the file.

Read File in Segments

This option enables you to read files in segments (chunks). Chunking files enables large files to be processed, one logical chunk at a time. A logical chunk (segment) is retrieved from a huge file, enabling the file to stay within memory constraints.

You can also read large XML files containing repeating elements and multiple namespaces. A use case is provided.

See [Read Large XML Files Containing Multiple Namespaces](#).

Note:

- If you select the **Read File in Segments** operation, you cannot specify an opaque or JSON schema. If you do, you receive the following runtime error:

NXSD has infinite loop.
 Flow has bad NXSD or bad input file which is causing infinite loop.
 Either NXSD is not designed properly or input file is not compatible with NXSD. Processing of file at stage read was not advancing and looping at same location in input file.
 Please download ICS flow and review NXSD file or inspect input file to ensure there are no infinite loop.
 Stage Read Failed

Table 5-2 - Read File in Segments

Property	Description
Configure File Reference	<ul style="list-style-type: none"> • Yes: Select to process an upstream operation that provides a file reference. • No: Select to process a file name.

Table 5-2 (Cont.) - Read File in Segments

Property	Description
Specify the File Reference (Appears if you select Yes for the Configure File Reference .)	Click the field and specify the file reference in any of the following ways: <ul style="list-style-type: none">• Entering manually.• Selecting from the list.• Clicking Switch to Developer View  to open the Sources tree.
Specify the File Name (Appears if you select No for the Configure File Reference .)	Click the field and specify a file name in any of the following ways: <ul style="list-style-type: none">• Entering manually.• Selecting from the list.• Clicking Switch to Developer View  to open the Sources tree.
Specify the Directory to read from	The Read File in Segments option creates a stage file action that includes a scope part. This enables you to drag actions inside the scope (such as for-each actions, additional stage file actions, and others) for more complex scenarios.
Segment Size	Click the field and specify a directory from which to read files in any of the following ways: <ul style="list-style-type: none">• Entering manually.• Selecting from the list.• Clicking Switch to Developer View  to open the Sources tree.
Sequential Processing	Enter the segment size. The segment size defaults to 200 records per segment, but can be changed. The minimum value is 200 records and the maximum value is 2000 records. Chunking files enables large files to be processed, one logical chunk at a time. A logical chunk (segment) is retrieved from a huge file, enabling the file to stay within memory constraints.
Remove Trailer	Select to read the segments sequentially.
Write File	
<p> Note:</p> <ul style="list-style-type: none"> • When writing to a stage file, there is no built-in support for headers. You must create headers by explicitly writing them to the file. • Multiple stage file action write operations appending data to the same stage file within a loop results in the data being overwritten. This results in data loss. 	

Table 5-3 - Write File

Property	Description
Specify the File Name	Click the field and specify a file name in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.  to open the Sources tree. <ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.
Specify the Output Directory	Click the field and specify an output directory in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.  to open the Sources tree. <ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.
Append to Existing File	Optionally select to append records to the existing file. Note: If you append JSON or XML content, the final file is invalid XML or JSON.
Encrypt	Click the check box, then select the public key of the target location to use to encrypt the file.

Zip Files

Table 5-4 - Zip Files

Property	Description
Specify the File Name	Click the field and specify a file name in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.  to open the Sources tree. <ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.
Specify the Directory to zip	Click the field and specify a directory to zip up in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.  to open the Sources tree. <ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.
Specify the Output Directory	Click the field and specify an output directory in which for the ZIP file in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.  to open the Sources tree. <ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.

Unzip File

Table 5-5 - Unzip File

Property	Description
Configure File Reference	<ul style="list-style-type: none"> • Yes: Select to process an upstream operation that provides a ZIP file reference. Once selected, you specify the file reference and the directory in which to unzip the file. • No: Select to process a ZIP file.
Specify the File Reference	Click the field and specify a file reference in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.
(Appears if you select Yes for the Configure File Reference .)	<ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.
Specify the Zip File Name	Click the field and specify a ZIP file name to read in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.
(Appears if you select No for the Configure File Reference .)	<ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.
Specify the Zip File Directory	Click the field and specify a directory in which to unzip the file in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.
	<ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.
Specify the Directory to Unzip	Click the field and specify the directory to unzip in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list.
	<ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.

Encrypt File

Table 5-6 - Encrypt File

Property	Description
Specify the File Reference	Click the field and specify a file reference in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. <ul style="list-style-type: none"> • Clicking Switch to Developer View  to open the Sources tree.

Table 5-6 (Cont.) - Encrypt File

Property	Description
Specify the File Name	<p>Click the field and specify a file name in any of the following ways:</p> <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. <p> • Clicking Switch to Developer View  to open the Sources tree.</p>
Specify the Output Directory	<p>Click the field and specify an output directory in any of the following ways:</p> <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. <p> • Clicking Switch to Developer View  to open the Sources tree.</p>
Specify PGP Key to encrypt file	<p>Select the private key of the target location to use to encrypt the file. You can encrypt a file up to 1 GB in size.</p> <p>When using the mapper, note that encrypt is visible as an element for mapping.</p>

Decrypt File

Table 5-7 - Decrypt File

Property	Description
Specify the File Reference	<p>Click the field and specify a file reference in any of the following ways:</p> <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. <p> • Clicking Switch to Developer View  to open the Sources tree.</p>
Specify the File Name	<p>Click the field and specify a file name in any of the following ways:</p> <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. <p> • Clicking Switch to Developer View  to open the Sources tree.</p>
Specify the Output Directory	<p>Click the field and specify an output directory in any of the following ways:</p> <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. <p> • Clicking Switch to Developer View  to open the Sources tree.</p>
Specify PGP Key to decrypt file	<p>Select the private key of the target location to use to decrypt the file. You can decrypt a file up to 1 GB in size.</p> <p>When using the mapper, note that decrypt is visible as an element for mapping.</p>

List File

Table 5-8 - List File

Property	Description
Specify the Directory to List Files from	Click the field and specify a directory from which to list files in any of the following ways: <ul style="list-style-type: none"> • Entering manually. • Selecting from the list. • Clicking Switch to Developer View  to open the Sources tree.
Specify the File Pattern to use	Specify the pattern of the file name to transfer to the output directory. Click the ? icon for the list of supported patterns.
List Files Recursively	Select to list the files recursively.

2. When complete, click > .

The Schema Options page is displayed if you selected a read or write stage file operation.

Define the Schema File

The Schema Options page enables you to define a schema. This page is displayed if you selected a read or write stage file operation.

1. Provide the following details.

Property	Description
Do you want to specify the structure for the contents of the file?	Select Yes to define a schema format to use for the files to transfer. Select No if a schema is not required and you want to send opaque files (for example, a GIF or PNG file).
Which one the following choices would be used to describe the structure of the file contents?	Select an option: <ul style="list-style-type: none"> • Sample delimited document (erg. CSV): Select to create a new schema from a CSV file. On a subsequent page of this panel, you are prompted to select the CSV file from which to create the schema. • XML schema (XSD) document: Select to create a new schema from an XML schema (XSD) file. You can upload a ZIP file with nested XML schema files. • Sample XML document (Single or No Name Space): Select to create a new schema from a sample XML file with a single or no name space. • Sample JSON document: Select to create a new schema from a JSON file. This option only appears if you selected the Read Entire File or Write File operation.

2. When complete, click > .

Define the Schema Format

Based on your selection on the Schema Options page, the Format Definition page enables you to select the file from which to create a schema. This page is displayed if you selected a read or write stage file operation.

1. Follow the instructions based on the schema option you selected:

 **Note:**

Oracle does not certify schemas that were manually updated with undocumented annotations. Use discretion if specifying these schemas. For example, runtime errors can occur if you use Oracle SOA Suite schemas that are annotated with constructs that are undocumented and not certified in Oracle Integration 3.

- **Sample delimited document (e.g. CSV)** ([Table 7](#))
- **XML schema (XSD) document** ([Table 8](#))
- **Sample XML document (Single or No NameSpace)** ([Table 9](#))
- **Sample JSON document** ([Table 10](#))

Table 5-9 - Sample Delimited Document (e.g. CSV)

Element	Description
• Select the Delimited Data File	Drag and drop the delimited comma-separated value (CSV) file from which to create the schema file. The content of the file is then displayed at the bottom of the page.
• Enter the Record Name	Enter the record name. This becomes the parent element in the created schema file for the record names selected as column headers from the CSV file.
• Enter the Recordset Name	Enter the recordset name. This becomes the root element of the created schema file.
• Select the Field Delimiter	Select one of the following supported file delimiter options: <ul style="list-style-type: none">• Single space• Comma• Semicolon• Tab• Pipe (for example, Name City Country)
• Character Set	Select a character set. The selected value is used as the encoding format while reading the sample data file.

Table 5-9 (Cont.) - Sample Delimited Document (e.g. CSV)

Element	Description
• Optionally Enclosed By	This value causes occurrences of the selected delimiter to be ignored during processing. For example, when processing the following record: Fred,"2 Old Street, Old Town,Manchester",20-08-1954,0161-499-1 718
• Use the First Row as Column Headers	If the selected Field Delimiter is "," and the Optionally Enclosed By value is <code>quot; ("</code>), then the value 2 Old Street, Old Town,Manchester is treated as a single record column.
• Detach	Displays by default the first row of the selected CSV file as the column headers.
• Mark All As Optional	Select to edit the CSV file in a separate window. Select to mark elements as optional in the schema file. By default, all elements are mandatory. You can also select the data type (for example, string, byte, integer, and so on) to use for each column in the table and mark specific elements as optional. While this option enables you to select all elements as optional, you must have at least one mandatory element to validate this page. This check box provides a convenient method to select the majority of elements as optional.

Table 5-10 - XML Schema (XSD) Document

Element	Description
• Select a New File	Drag and drop an existing XML schema file or schema archive file from the file system.
• Selected File Name	Displays the selected schema file name.
• Select the Element Name	Select the schema element. This field is displayed after the XML schema file is selected. The element name is treated as the root element in the uploaded schema file.
• Select Repeating Batch Element	Click the field and specify the repeating node in the schema to support payload chunking in any of the following ways: <ul style="list-style-type: none">• Entering manually.• Selecting from the list. • Clicking Switch to Developer View  to open the Sources tree.

Table 5-11 - Sample XML Document (Single or No Namespace)

Element	Description
• Select a New File	Drag and drop a sample XML document from the file system. The file must contain no namespace or a single namespace.
• Selected File Name	Displays the selected schema file name.
• Select the Schema Element	Select the schema element. This field is displayed after the sample XML file is selected. The element name is treated as the root element in the uploaded schema file.
• Select Repeating Batch Element	Click the field and specify the repeating node in the schema to support payload chunking in any of the following ways: <ul style="list-style-type: none">• Entering manually.• Selecting from the list.
	• Clicking Switch to Developer View  to open the Sources tree.

Table 5-12 - Sample JSON document

Element	Description
• Select a New File	Drag and drop a sample JSON document from the file system.
• Selected File Name	Displays the selected schema file name.
• Select the Schema Element	Select the schema element. This field is displayed after the JSON file is selected. The element name is treated as the root element in the uploaded schema file.

2. Complete the fields, and click **>**.
3. Review your selections on the Summary page, and click **Done**.

Review the Stage File Action in the Integration Canvas

Once design is complete, the stage file action is displayed in the integration canvas.

- If you designed a stage file action with a **Read File in Segments** operation, a scope portion is created. Place your cursor over the icon to display a message indicating that segmentation (chunking) is enabled.

You can drag additional actions into the scope part of the stage file action to perform tasks. For example, you can add a for-each action to the stage file action to process the segmented chunks of large files one record at a time. You can also add child stage file actions to perform further processing on each of the chunks. However, you cannot configure additional chunking on the child stage file action.



 **Note:**

If a variable is declared outside of a loop of a stage file action that was configured with the **Read File in Segments** operation and updated within the stage file action loop, the last updated value for that variable is available outside of the loop. In such a case, the DOP (degree of parallelism) is set to 1 to avoid concurrency issues.

- If you designed a stage file action with an operation other than a **Read File in Segments** operation, a scope portion is not included.

Complete your integration design and configure any business identifiers for tracking fields in messages (including file storage-related parameters).

Related Documentation

The following sections provide examples of using a stage file action in an integration:

- Create an Integration to Import and Process Bulk Files in *Using the FTP Adapter with Oracle Integration 3*
- Process RaaS Reports that Support the Delivery of Data in CSV Format in *Using the Workday Adapter with Oracle Integration 3*
- Produce Messages to an Apache Kafka Topic in *Using the Apache Kafka Adapter with Oracle Integration 3*
- Use Bulk Response Operations in an Integration in *Using the Salesforce Adapter with Oracle Integration 3*

Receive Requests for Multiple Resources in a Single REST Adapter Trigger with a Pick Action

You can expose multiple entry points to a single integration that uses the REST Adapter as the trigger connection. Each entry point can be configured with a different resource URI and HTTP action/verb, when necessary. This feature eliminates the need to create multiple integrations (each with a separate resource URI and verb) to perform different operations.

This feature is useful in the following scenarios:

- Supports multiple resource URIs and actions/verbs in a specific integration through the REST Adapter.
- Supports multiple integration patterns (for example, synchronous and asynchronous). The combination of a resource and a verb is called an operation. Each operation can be configured as a synchronous or an asynchronous one-way service. Asynchronous services accept the request and return immediately with an HTTP 202 status.

 **Note:**

This feature is only available when creating an application integration.

1. Drag a REST Adapter into the canvas as a trigger connection of an application integration.

This invokes the Adapter Endpoint Configuration Wizard.

2. On the Basic Info page, specify the following:
 - a. Enter a name and optional description.
 - b. Select the **Select to configure multiple resources or verbs (maximum 11)** check box.
3. On the Resource Configuration page, specify the operation name, the endpoint relative resource URI, and the action/verb to perform, then click **Next**.

The screenshot shows the 'Resource Configuration' page of the Adapter Endpoint Configuration Wizard. It has the following fields:

- Provide an operation name ***: A text input field containing "modifyEmployee".
- What does this operation do?**: A text input field containing "Describe what this operation does".
- What is the endpoint's relative resource URI? ***: A text input field containing "/emp".
- What action do you want to perform on the endpoint? ***: A dropdown menu showing "PATCH".
- Based on your selections, you can add parameters or configure a request and/or response for this endpoint.**: A note at the bottom of the configuration area.
- Select any options that you want to configure:** A section with three checkboxes:
 - Add and review parameters for this endpoint
 - Configure a request payload for this endpoint
 - Configure this endpoint to receive the response

4. On the Operations page, click **Add another operation** to add another resource URI and action/verb, then click **Next**.

You are returned to the Resource Configuration page.
5. On the Resource Configuration page, specify a different operation name, endpoint relative resource URI, and action/verb to perform, then click **Next**.
6. Continue this process to add as many operations as necessary.
7. On the Operations page, note that all operations are displayed. Each operation represents a different branch in a single integration.

The screenshot shows the REST adapter configuration interface. At the top, there are tabs: Request Headers (5), CORS Config... (6), Response (7), Response Headers (8), and Operations (9). The Operations tab is selected. Below the tabs, the title "Operations" is displayed, followed by the message "Operation modifyEmployee edited". A table lists six operations:

Operation	Resource	HTTP Method	Edit/Delete
modifyEmployee	/emp	PATCH	<input type="checkbox"/>
updateEmployee	/emp	PUT	<input type="checkbox"/>
getEmployee	/emp	GET	<input type="checkbox"/>
getEmployeeLocation	/emp/loc	GET	<input type="checkbox"/>
deleteEmployee	/emp	DELETE	<input type="checkbox"/>
addEmployee	/emp	POST	<input type="checkbox"/>

* Double click to edit table cells and hit Enter/Return key to commit changes

Add another operation

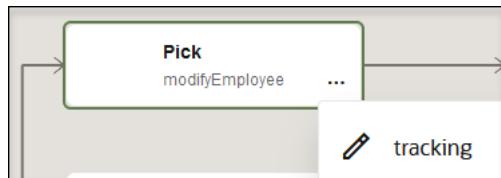
- On the Summary page, review your selections, and click **Done**.

For this example, there are six branches (**modifyEmployee**, **updateEmployee**, **getEmployee**, **getEmployeeLocation**, **deleteEmployee**, and **addEmployee**).

- Place your cursor over each branch to show details about the operation/resource name, action/verb, and resource URI.
- Further configure your integration by dragging appropriate actions *after* each operation name. You cannot add actions between the REST Adapter icon and the operation name.



11. Select **Actions ...**, then **tracking** for the pick action operation name for which to define business identifier tracking variables.



12. Set the tracking variables for each pick action operation branch. Tracking variables are populated per the operation. You must set primary tracking variable for all operations to activate the integration. If any branches are not configured with tracking variables, a red error icon is displayed on the node. Place your cursor over the error icon to show a detailed error message.
13. Activate and invoke the integration to take the appropriate branch.

Once activated, the adapter metadata page displays information corresponding to each of the configured resources and verbs.

The swagger displays all the configured resources and the verbs for each such resource. Similar to swagger from a single resource REST trigger endpoint, you can use this swagger in another invoke connection to automatically discover and browse the resource implemented.

 **Note:**

You can edit the action in the integration to remove or add operation branches. This enables you to convert to and from a standard trigger-based integration with a single operation to an integration with multiple operation branches.

Log Messages with a Logger Action

You can log messages to the activity stream and diagnostic logs by adding the logger action at any point in the integration. You create a log message in the logger action that is a static message or variable-populated message.

Create a Logger Action

To create a logger action:

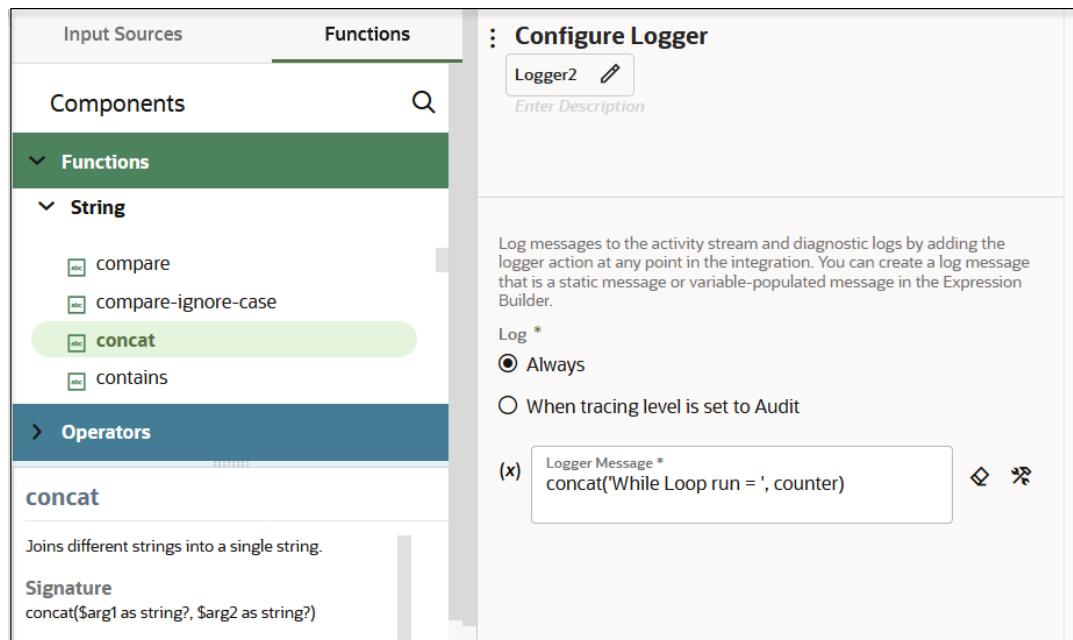
1. Add a logger action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Log** action to the appropriate location.
- Click  at the location where you want to add the log action, then select **Log**.

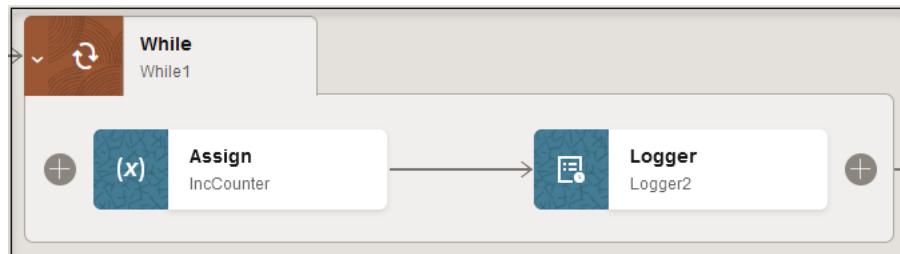
The Configure Logger panel opens.

2. Click  under **Configure Logger** to enter a name and optional description for the log action.
3. Select whether to always log this message or log it only when tracing is set to audit. You can enable tracing when you activate an integration. See [Activate an Integration](#).

4. Specify a static or variable-populated message to be displayed in the activity stream and diagnostic logs. Drag elements from the **Sources** tree to create a message.



5. When complete, click **Save**. For this example, a log action is included in the integration.



Track the Status of a Logger Action During Runtime

During runtime, the messages in the loggers are written to the activity stream and diagnostic logs.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Click the integration instance.
3. Scroll through the activity stream.
4. Select **Download** to download the logs.
5. Open the zip file and view the log messages you created.

Process Tasks in Parallel with a Parallel Action

You can use a parallel action to process tasks in parallel to improve integration performance and response times. A parallel action allows the path of an integration to be split into multiple

branches. Each branch is processed in parallel due to their independence from each other. Messages are sent to each service endpoint in parallel.

When all tasks are completed, all branches are automatically synchronized and merged at their termination points in the parallel action and the main path of the integration is resumed. Any variables created inside the parallel branches (for example, invoke responses) are available for use in the main path.

- [Use Cases](#)
- [Guidelines and Restrictions](#)
- [Design Time Behavior](#)
- [Runtime Behavior](#)

Want to see a video demo of the parallel action?



Use Cases

A parallel action is useful for use cases such as the following:

- An order processing integration contains individual tasks for checking inventory, customer credit, and legal constraints. These tasks are processed in parallel to ensure a better response time.
- An integration obtains the lowest price for an item from suppliers. The integration checks three suppliers for prices and takes the best offer. The three price checks are done in parallel because they are independent of each other. Once the three suppliers have responded, the branches are terminated. The three prices are compared to identify the lowest. The integration then continues its main path by placing the order with that supplier.
- An integration processes an incoming order from both an inventory and legal perspective (for example, can the product be shipped to the customer's country). Both checks are done in parallel.

Guidelines and Restrictions

- You can create a maximum of five branches to a parallel action. If you attempt to create a sixth branch by selecting **Actions ...**, and then **Add** on the parallel action, the selection is disabled.
- There is no limit to the number of parallel actions you can include in an integration.
- Synchronous and asynchronous integrations are supported.
- You cannot add a parallel action inside another parallel action.
- Parallel actions can be used in application integrations and schedule integrations.
- Only map, scope, switch, logger, JavaScript, B2B, and notification actions and invoke connections are supported in a parallel action branch.
- You can add parallel actions to the branches of a pick action.
- You can add parallel actions inside scope and switch actions.
- Nesting of parallel actions in for-each actions, while actions, or stage file actions with chunking is not supported.
- No independent branch-specific fault handling is supported. You can catch faults at the integration level with a global fault handler.
- Synchronization of global variables is not supported.

- Branch-specific timeouts are not supported.

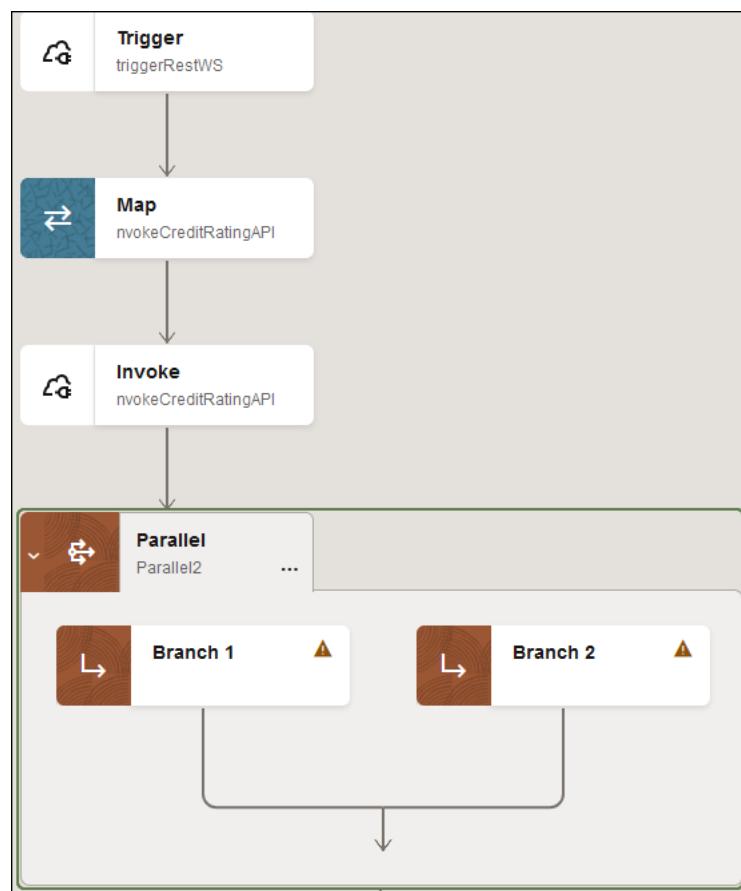
Design Time Behavior

The following example provides a high-level overview of how to design a specific parallel action. This is just one example of parallel action use.

1. Add a parallel action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Parallel** action to the appropriate location.
- Click  at the location where you want to add the parallel action, then select **Parallel**.

A parallel action with two branches is created.



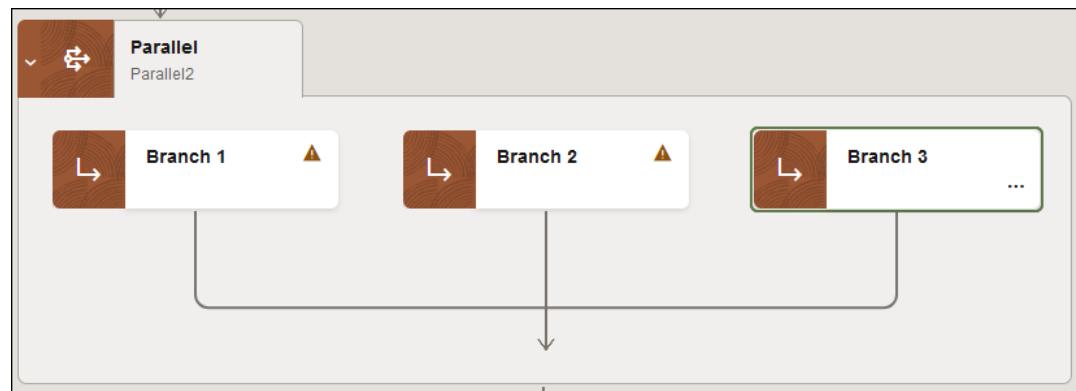
2. Select **Actions ...** to perform the following tasks.

- **Edit**: Change the name of the parallel action.
- **Add**: Add more branches. You can add a maximum of five branches.
- **Delete**: Delete the parallel action.
- **Collapse/Expand**: Collapse or expand the parallel action.
- **Cut**: Cut the parallel action and paste it elsewhere in the integration. You can also cut and paste existing maps/invokes into the new parallel branches to make them process

in parallel. You *cannot* cut an individual branch of a parallel action and paste it into another parallel action.

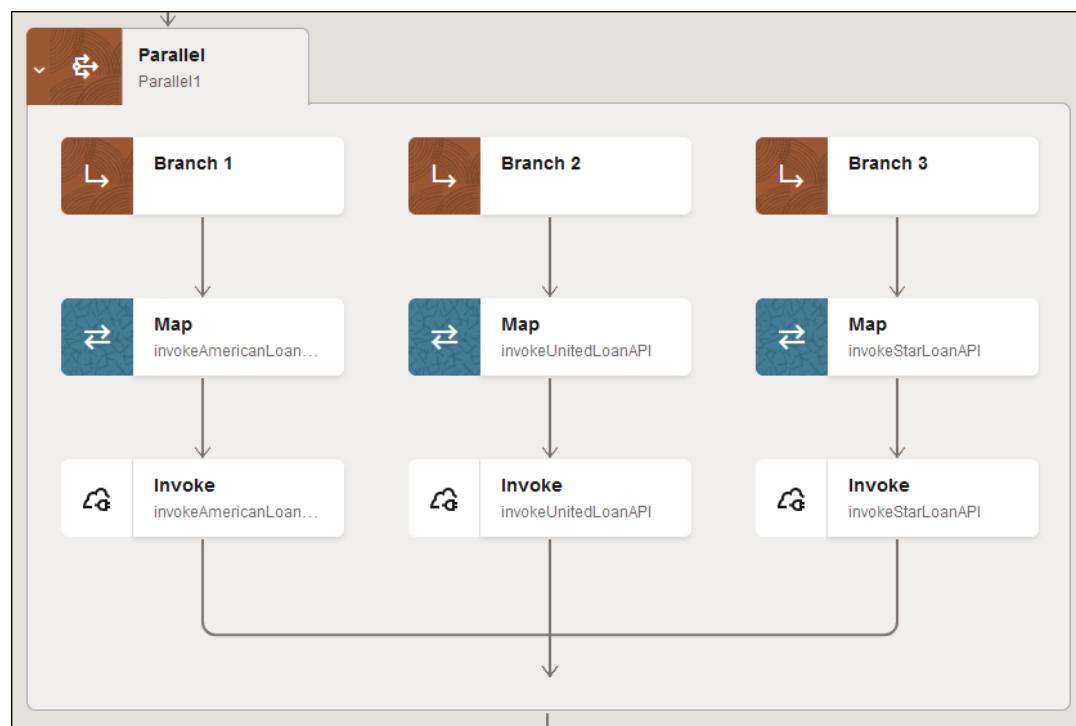
See [Cut and Paste Elements in the Integration Canvas](#).

For this example, a third branch is added.



For this example, the three branches are designed to obtain loan offers from three service endpoints and then select the best one.

3. Add and configure three REST Adapter connections to invoke the following service endpoints. At runtime, all three loan service endpoints are invoked in parallel to identify their specific loan offers.
 - American Loan
 - United Loan
 - Star Loan
4. In the mapper for each invoke action, map your SSN to the loan application for that service endpoint.

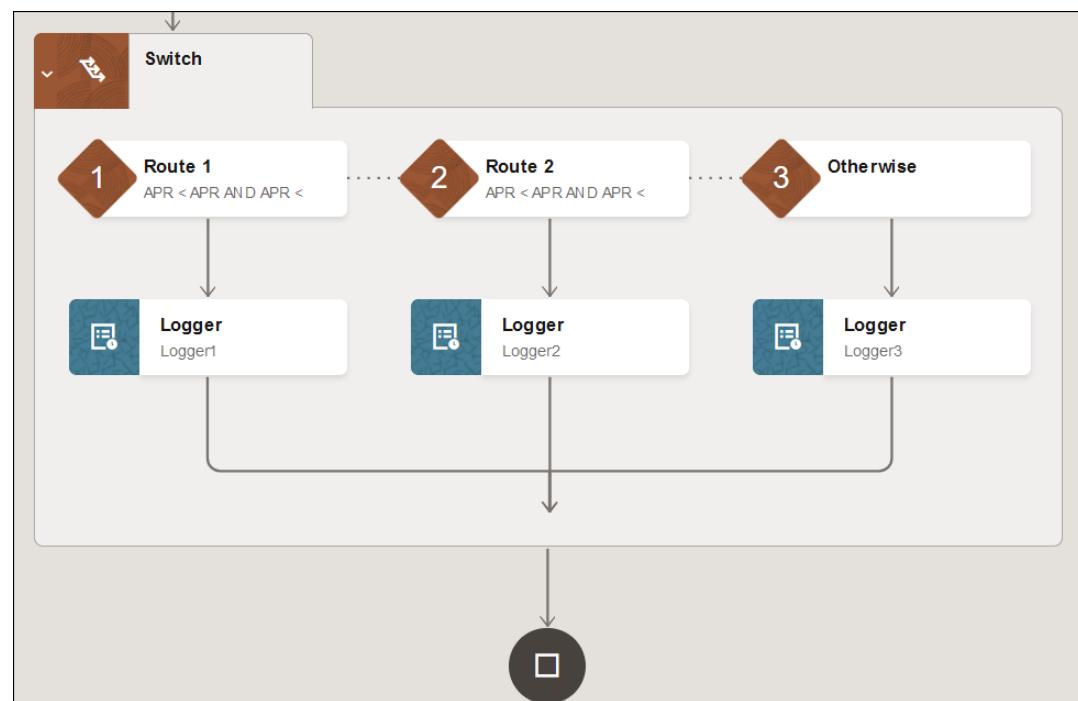
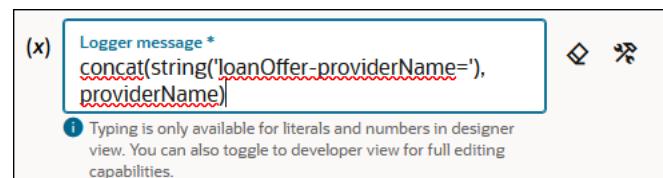


At the bottom, the three branches are synchronized at their termination points and the parallel action rejoins the main branch of the integration. The main integration does not continue until all branches are complete

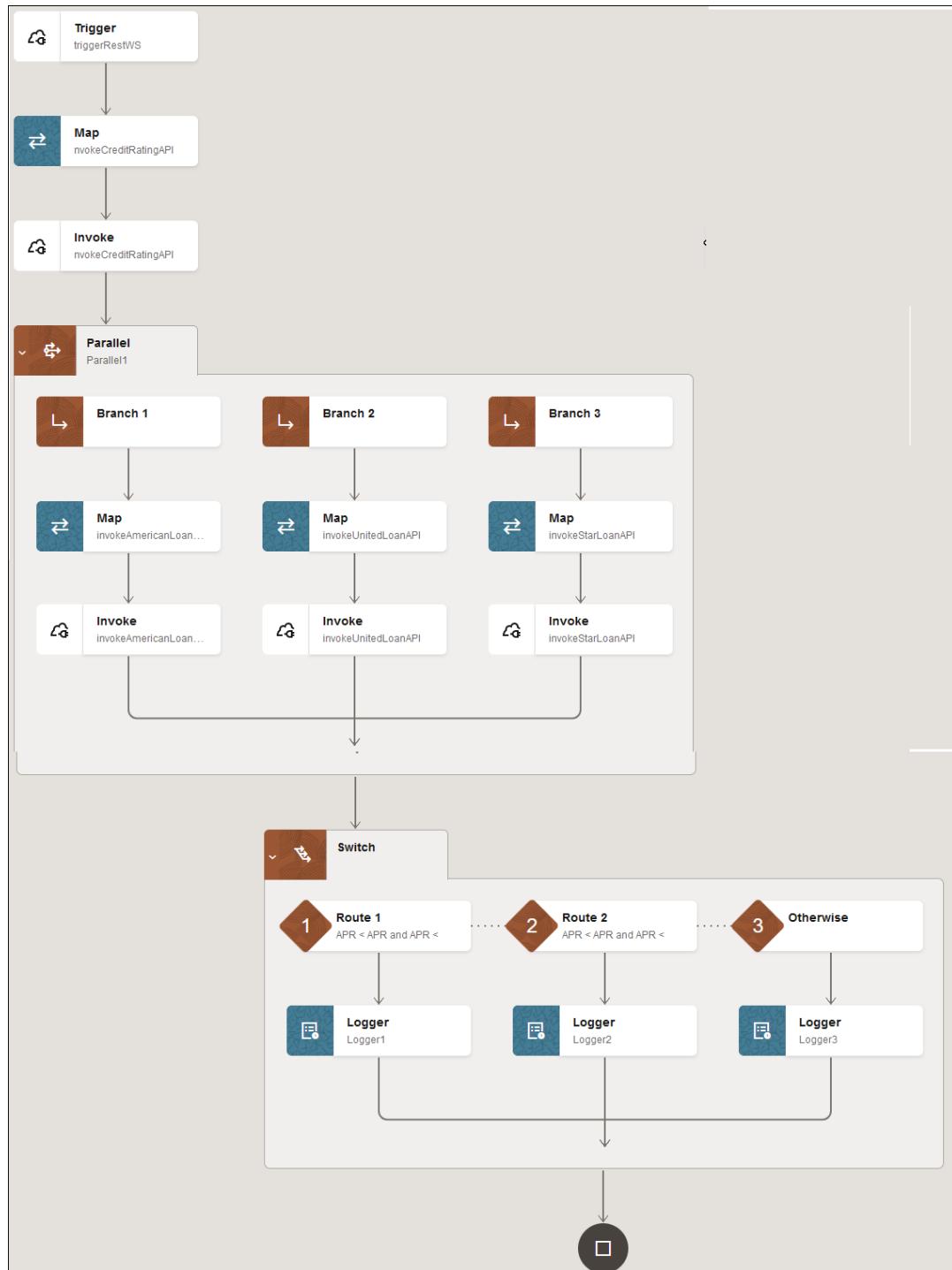
- Add a switch action to define the routing rules to check for the best offer returned from the loan service endpoints.



- Define the message in a log action for each switch branch.



When complete, the integration looks as follows.

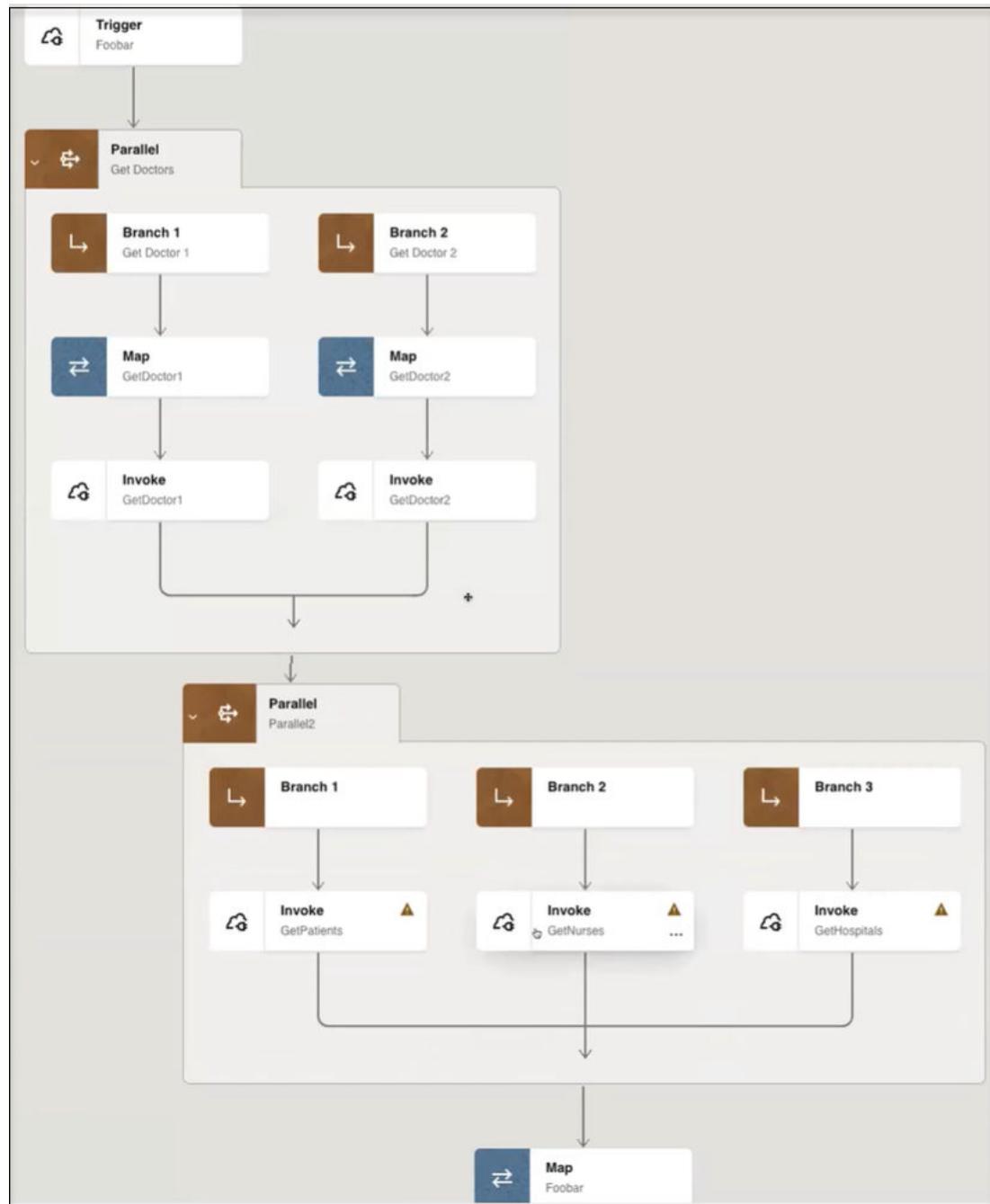


The integration is now ready to be activated and invoked.

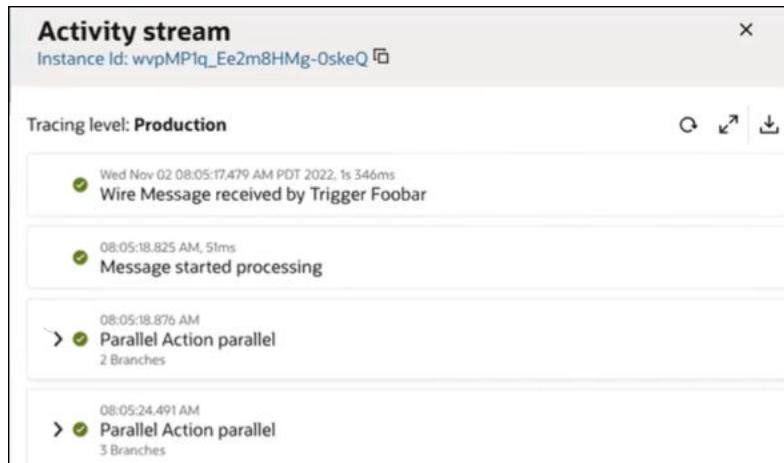
Runtime Behavior

You can view the status of parallel branches in the activity stream during runtime. For example, assume your integration is designed with the following:

- One parallel action with two branches
- A second parallel action with three branches



1. Invoke an instance of your integration and go to the Instances page. During runtime, the activity stream shows both parallel actions and their number of branches. For this example, both parallel actions processed successfully.



2. Expand each branch to view more specific details.

Invoke Oracle Cloud Infrastructure Functions Directly from an Integration with an OCI Function Action

Oracle Integration provides native support for invoking Oracle Cloud Infrastructure functions. This support enables you to invoke Oracle Cloud Infrastructure functions with an OCI Function action in an integration.

Oracle Cloud Infrastructure functions are part of a fully managed, multi-tenant, highly scalable, on-demand, Functions-as-a-Service platform. Use Oracle Cloud Infrastructure functions to focus on writing code to meet business needs. See [Overview of Functions](#) and [Functions QuickStart Guides](#).

Note:

Ensure that your cloud tenancy uses identity domains. The OCI Function action in the integration canvas does *not* work in cloud tenancies that are not enabled for identity domains.

- [Capabilities](#)
- [Prerequisites](#)
- [Invoke Oracle Cloud Infrastructure Functions](#)

Want to see a video demo of the OCI Function action?



Capabilities

Oracle Cloud Infrastructure functions provide serverless execution of functions within Oracle Integration to process complex logic not easily performed in Oracle Integration, such as the following:

- Creating a digital signature based on a key against a specific payload.
- Supporting a custom ZIP format such as GZIP or TAR.

- Taking and parsing a PDF file for data fields that can then participate in an integration.

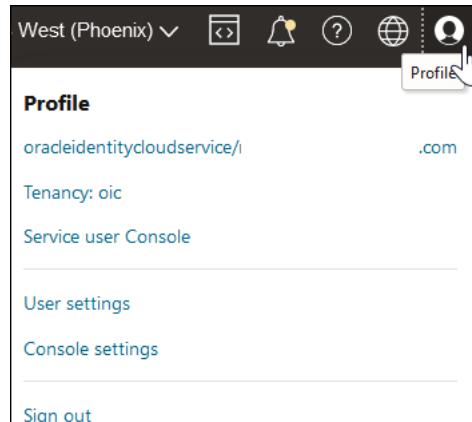
Oracle Integration supports whatever functions you create using Oracle Cloud Infrastructure functions. Oracle Cloud Infrastructure also provides a catalog of prebuilt functions. See [Creating Functions Using Pre-Built Functions](#).

Prerequisites

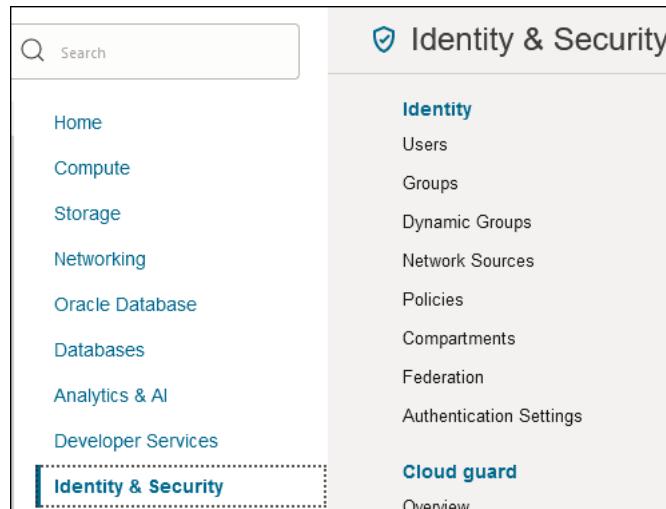
You must perform the following prerequisites from the Oracle Cloud Infrastructure Console before you can invoke functions from an integration. Not performing these prerequisites results in the following error:

An authentication issue was encountered when invoking Function. Please check if the prerequisites for calling OCI services have been properly setup for this service instance. OCI service integration requires dynamic groups and policies granting access to be created in OCI Console. Please consult the documentation for more information.

1. Ensure that your cloud tenancy uses identity domains. Invoking Oracle Cloud Infrastructure functions from an integration does *not* work in cloud tenancies that are not enabled for identity domains. If you are unsure, ask your administrator for details. Your cloud tenancy does *not* use identity domains if you observe either of the following:
 - In the upper right corner, you select your **Profiles** icon and don't see an entry for identity domain.



- From the **≡** menu, you select **Identity & Security** and don't see **Domains** under the **Identity** section.



2. Set up your environment to invoke Oracle Cloud Infrastructure functions from an integration. See [Functions QuickStart on Cloud Shell](#). This section describes how to set up the following:
 - Set up your tenancy:
 - Create groups and users (if not already done so).
 - Create a compartment in which to create functions. You specify this compartment when configuring the **OCI Function** action in the integration canvas.
 - Create a virtual cloud network (VCN) and subnets. For example:

Name	State	IPv4 CIDR Block	IPv6 Prefix	Default Route Table	DNS Domain Name	Created
oci-native-vcn	Available	10.0.0.0/16	-	default_route_table for oci-native-vcn	oci	Fri, Feb 17, 2023, 19:03:46 UTC

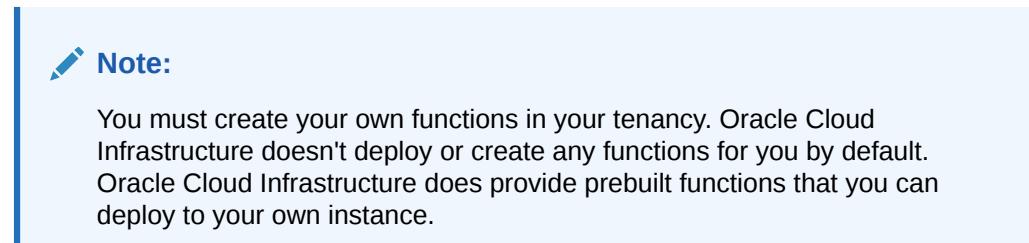
- Create a policy for groups and services. This policy enables you to create functions. For example:

Name	Description	Statements	Created
cbf-zip-object_policy	Autogenerated policies for the pbf-zip-object function created from pre-built function. user: icm_admin Tenant: ocicn	4	Mon, Apr 10, 2023, 23:12:59 UTC
object-storage_PBF_DEMO_policy	Autogenerated policies for the object-storage_PBF_DEMO function created from pre-built function. user: icm_admin Tenant: ocicn	4	Thu, Mar 30, 2023, 21:33:06 UTC

- Create an application in which to create the functions to invoke. You specify this application when configuring the **OCI Function** action in the integration canvas. For example:

Applications in oic-oci-native-actions-testing Compartment					
Applications		Create application			
Pre-Built Functions		Name	Status	Created	Last updated
knowledge-transfer-demo		knowledge-transfer-demo	ACTIVE	Thu, Mar 30, 2023, 21:32:25 UTC	Thu, Mar 30, 2023, 21:32:25 UTC
oci-oci-native-actions-testing	oci-oci-native-functions	TestCreateAppDeleteThis	ACTIVE	Thu, Feb 23, 2023, 18:35:02 UTC	Thu, Feb 23, 2023, 18:35:02 UTC
oci-oci-native-functions		TestCreateApp	ACTIVE	Wed, Feb 22, 2023, 17:57:56 UTC	Wed, Feb 22, 2023, 17:57:56 UTC
		oci-oci-native-functions	ACTIVE	Fri, Feb 17, 2023, 19:11:05 UTC	Fri, Feb 17, 2023, 19:11:05 UTC

- Set up your Cloud Shell development environment.
 - Create, deploy, and invoke your function. For example:



Functions » Applications » oic-oci-native-functions

oic-oci-native-functions

[Move application](#) [Add tags](#) [Delete](#)

[Application information](#) [Tags](#)

General information

OCID: `...nicj7ra` [Show](#) [Copy](#)
Compartment: oic-oci-native-actions-testing
Logging policy: OCI Logging
Trace name: None
Created: Fri, Feb 17, 2023, 19:11:05 UTC
Last updated: Fri, Feb 17, 2023, 19:11:05 UTC
Signature verification: Disabled

Network information

Subnets: [public](#)
Network security groups: None [Add](#)

Resources

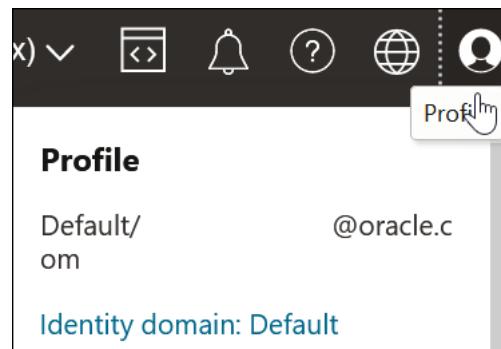
[Getting started](#)[Functions](#)[Configuration](#)[Metrics](#)[Logs](#)[Traces](#)[Tag filters](#) [add](#) [clear](#)

Functions

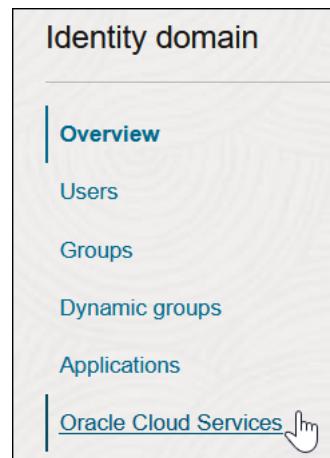
[Create function](#)

Name	Image	Image digest	Invoke endpoint	Provisioned concurrency units	Last updated
<code>create-thumbail</code>	<code>phx.oci.oracleoci/test-image-repo/</code>	<code>sha256:2526bf5ac6d09cc5cb6c73cdcf05d84ba87f08cb6616d</code>	<code>https://oedtbf09ja767ra/</code>	-	Fri, Feb 17, 2023, 19:11:05 UTC
<code>asyncfn</code>	<code>phx.oci.oracleoci/test-image-repo/</code>	<code>sha256:4e6bd699a912c12d1a3a7</code>	<code>https://18a0d181c1c351cf7d9dbfe489c</code>	-	Fri, Feb 17, 2023, 19:11:05 UTC
<code>hello-world</code>	<code>phx.oci.oracleoci/test-image-repo/</code>	<code>sha256:4899bebe340ccf9718c456</code>	<code>https://i63c9d3774ef77f795953ec92e76</code>	n -	Fri, Feb 17, 2023, 19:11:05 UTC
<code>hello-python</code>	<code>phx.oci.oracleoci/test-image-repo/</code>	<code>sha256:4899bebe340ccf9718c456</code>	<code>https://i63c9d3774ef77f795953ec92e76</code>	n -	Fri, Feb 17, 2023, 19:11:05 UTC

3. Create a dynamic group.
 - a. Obtain the client ID of the OAuth application for the Oracle Integration instance.
 - b. In the upper right corner, select **Profile**, then click the identity domain.



- c. In the left navigation pane, click **Oracle Cloud Services**.



The Oracle Cloud Services page for your domain appears.

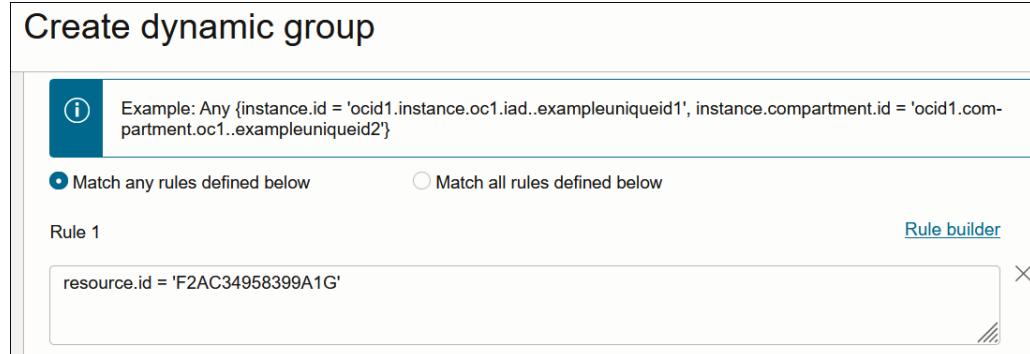
- d. In the **Name** column, click your service instance.
- e. Scroll down to the **General Information** section and copy the client ID value to use to create your dynamic group.
4. Scroll to the breadcrumbs at the top and click **Default domain**.



5. In the left navigation pane, click **Dynamic groups**.
6. Click **Create Dynamic Group**.
7. Enter the following details:
 - a. In the **Name** and **Description** fields, enter values. These fields are required.

- b. In the **Matching Rules** section, enter the required rule. The resource ID you specify must match the client ID of the OAuth application of your Oracle Integration instance. Ensure that you enclose the value in single quotes. For example:

```
resource.id = 'client_ID'
```



8. Assign a policy to the dynamic group to enable you to invoke Oracle Cloud Infrastructure functions from an integration.
9. Scroll to the breadcrumbs at the top and click **Identity**.



10. In the left navigation pane, click **Policies**.
11. Click **Create Policy**.
12. Select the compartment in which to create the policy.
13. Enter the following details:
 - a. In the **Name** and **Description** fields, enter values. These fields are required.
 - b. In the **Policy Builder** section, build the required policy for the dynamic group. The following example allows the dynamic group to use the entire functions family, but you can set more granular policies, as needed. See [Controlling Access to Invoke and Manage Functions](#) and [Policy Statements to Give OCI Functions Users Access to Function-Related Resources](#).

Allow dynamic-group *group_name* to use functions-family in compartment *compartment_name*

Where:

- *group_name*: Is the dynamic group name you created.
- *compartment_name*: Is the compartment where the functions to invoke are hosted.

This enables the Oracle Integration instance associated with the dynamic group to invoke Oracle Cloud Infrastructure functions with the **OCI Function** action in an integration.

Invoke Oracle Cloud Infrastructure Functions

This section provides an overview of how to configure an **OCI Function** action in the integration canvas to invoke the Oracle Cloud Infrastructure functions available in your application.

1. Add an **OCI Function** action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **OCI Function** action to the appropriate location.
- Click  at the location where you want to add the parallel action, then select **OCI Function**.

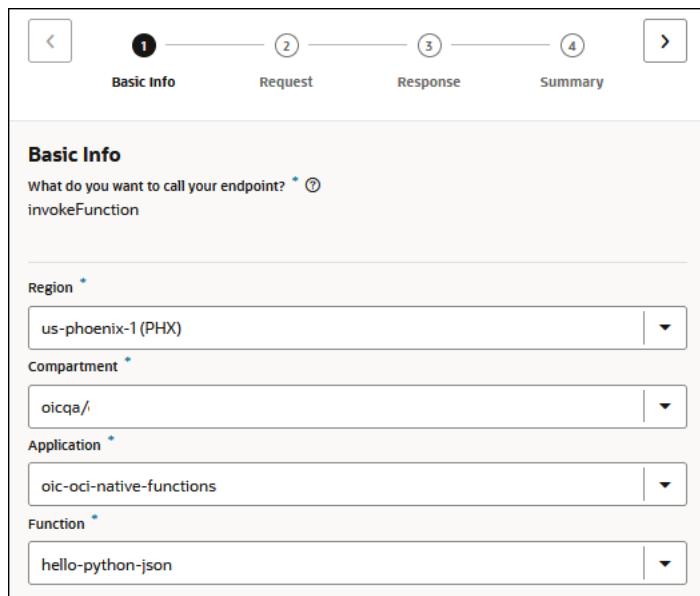
 **Note:**

If your cloud tenancy is not using identity domains, an error message appears and the page is blank.

The OCI Functions wizard appears.

2. On the Basic Info page, enter the following information to identify the function to invoke, then click **>**. When you enter details in a field, the page refreshes to show more fields.

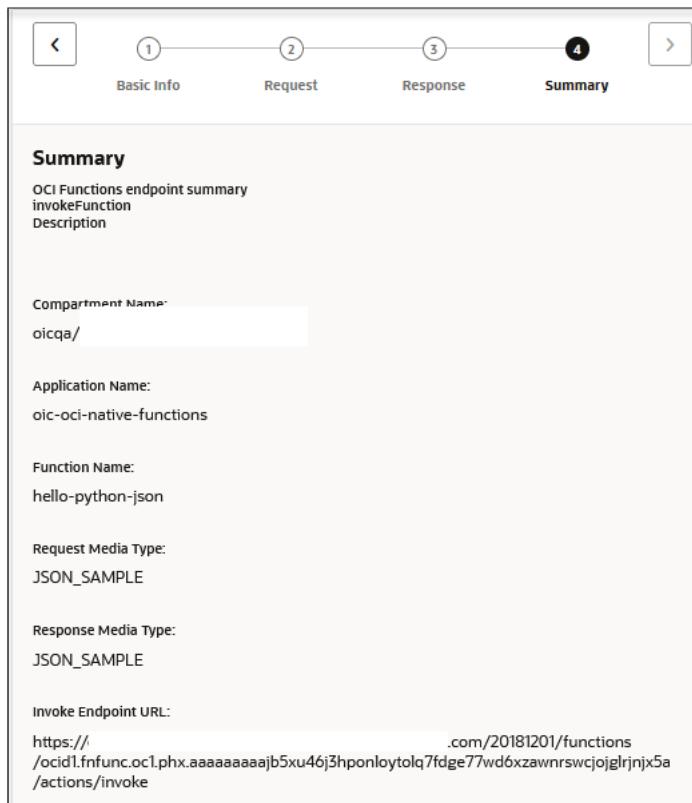
Element	Description
Region	Select the region where the functions are deployed.
Compartment	Select the compartment where the functions to invoke are hosted.
Application	Select the application. Oracle Cloud Infrastructure functions are organized in sets called applications. An application can contain multiple functions.
Function	Select the Oracle Cloud Infrastructure function to invoke. The functions you are entitled to invoke by your application are displayed for selection.



3. On the Request page, select the payload type and specify the content, then click >.
 - **JSON Sample:** Drag and drop or select a file, or manually enter the payload.
 - **XML Schema:** Drag and drop or select a file.
 - **XML Sample (Single or No NameSpace):** Drag and drop or select a file.
 - **Binary:**
 - For files, simply select **Binary** and map the content.
 - For plain/text request/response payloads, select **Binary** and perform the following encoding:
 - * To provide text as input, you must first encode the text as Base64, then decode that Base64 to a stream reference.

`decodeBase64ToReference (encodeBase64 (Name))`
 - * To receive the text output, you need to encode the output stream reference to Base64, then decode that Base64 to text.

`decodeBase64 (encodeReferenceToBase64 (Stream Reference))`
4. On the Response page, select the response payload type and specify the content, then click >. The available response payload types are the same as the request payload types.
5. Review your selections on the Summary page. For this example, the **hello-python-json** function is invoked. The request and response media is in JSON format.



6. Click **Done**.

Publish Events in an Integration with a Publish Event Action

You can publish an event in an integration. Subscription integrations can then subscribe to this event.

1. Add a **Publish Event** action to an integration in either of the following ways:
 - On the side of the canvas, click **Actions**  and drag the **Publish Event** action to the appropriate location.
 - Click  at the location where you want to add the publish event action, then select **Publish Event**.

The Choose event panel opens.
2. Specify an event in either of two ways:
 - a. Select the event, then click **Choose**. If you are unsure of the event contents, click .
 - b. Click **Define new event** to create a new event. See [Create Events to Publish and Subscribe to in Integrations](#).
3. Review your selections, then click **Finish**.

See [Publish Events in an Integration](#).

Add JavaScript Functions to an Integration with a JavaScript Action

You can add JavaScript functions to the integration.

Create a JavaScript Action

Note:

Note the following restrictions when using the JavaScript action:

- The JavaScript action has a timeout threshold of 15 seconds. Any JavaScript function that processes for more than 15 seconds fails with a timeout error.
- JavaScript functions are not allowed to make outbound calls to external services. Any outbound calls are blocked and ultimately fail.
- Network, disk access, or thread access functions are not supported.

1. Add a Javascript action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Javascript** action to the appropriate location.
- Click  at the location where you want to add the Javascript action, then select **Javascript**.

2. Click **Edit**  under **Configure Javascript** to enter a name and optional description for the JavaScript action.

3. Click the **Function** button.

The **Functions List** appears.

4. Select a function.

The function and **Sources** tree are displayed. Function details are shown, including the input and output parameters.

The screenshot shows the 'Configure Javascript' interface. On the left, the 'Sources' tree is expanded, showing nodes like 'add *', 'x *', 'y *', '\$self', and tracking variables. The 'Function: add' configuration panel is displayed on the right, showing the function details: 'Javascript Library: JSAddFunction - 01.00.0000', 'Output Description: Result', and 'Output Parameters: retValue (double)'. Below this, the 'Input Parameters:' section is shown with two fields: 'param1:' and 'param2:'. A 'Replace' button is located in the top right corner of the configuration panel.

- Click the first input parameter to open a **Value** field.

A close-up view of the 'Input Parameters:' dialog. It shows a single input field labeled 'param1:' with a placeholder '(x)' and a 'Value' field containing the value 'x'. There are also icons for copy and paste.

- Drag a value from the **Sources** tree.

The screenshot shows the 'Configure Javascript' interface again. The 'Sources' tree on the left has the 'x *' node highlighted with a green background. The 'Function: add' configuration panel on the right remains the same as in the previous screenshot, with the 'param1:' field set to 'x'. The 'Replace' button is visible in the top right of the configuration panel.

7. Perform the same steps to add a value for the second input parameter.
8. Click **Save**.

Track the Status of a Javascript Action During Runtime

During runtime, you can track the status of the JavaScript action on the Instances page through the tracking diagram and activity stream for an activated integration. This is only possible if there is a tracking instance.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Click the business identifier value of the integration to track.
3. The integration (including any JavaScript actions) is displayed. Any JavaScript action failures are identified by red.

Details about processing status (including any JavaScript actions) are displayed, including any failures.

See [Use Libraries to Manage Integration Functions](#).

Invoke Oracle Cloud Infrastructure Object Storage from an Integration with an OCI Object Storage Action

You can natively invoke Oracle Cloud Infrastructure Object Storage from an integration without the need to configure an explicit REST Adapter connection.

- [Oracle Cloud Infrastructure Object Storage Concepts](#)
- [Prerequisites](#)
- [Invoke Oracle Cloud Infrastructure Object Storage from an Integration](#)

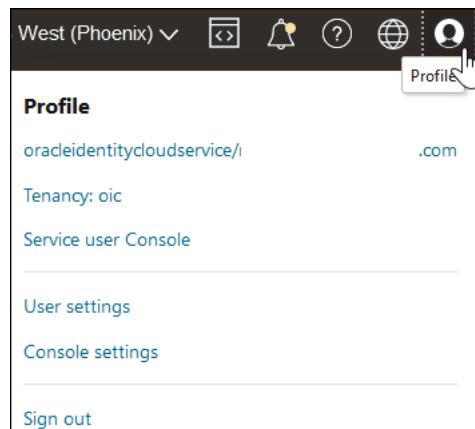
Oracle Cloud Infrastructure Object Storage Concepts

Oracle Cloud Infrastructure Object Storage is an internet-scale, high-performance storage platform that offers reliable and cost-efficient data durability. Oracle Cloud Infrastructure Object Storage enables you to securely store any type of data in its native format. An unlimited amount of unstructured data of any content type is supported, including analytic data and rich content such as images and videos. See [Object Storage](#) and [Overview of Object Storage](#).

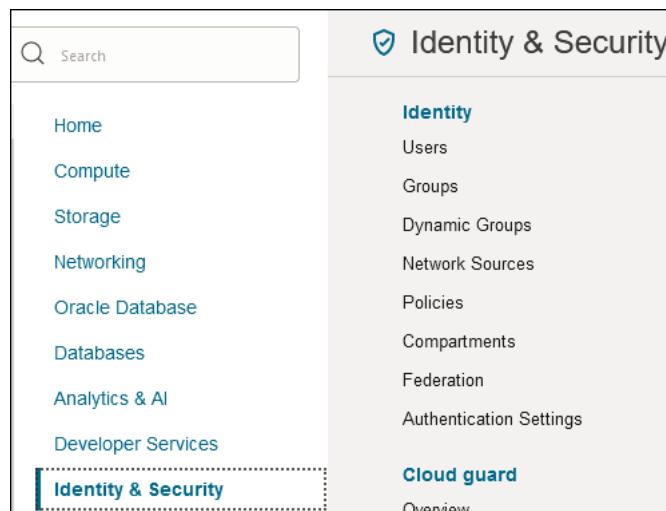
Prerequisites

To natively invoke Oracle Cloud Infrastructure Object Storage, you must satisfy the following Resource Principal Session Token (RPST) prerequisites in the Oracle Cloud Infrastructure Console.

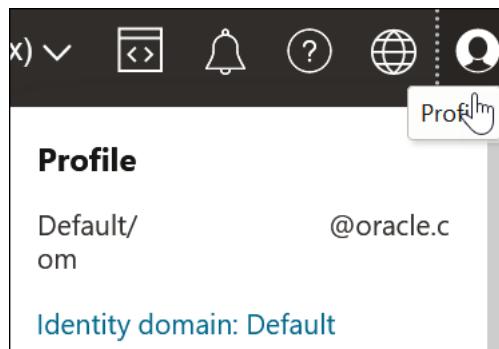
- Create a dynamic group
- Create a policy to grant access to Oracle Cloud Infrastructure Object Storage.
- Ensure that your cloud tenancy uses identity domains. Your cloud tenancy does *not* use identity domains if you observe either of the following:
 - In the upper right corner, you select your **Profiles** icon and don't see an entry for identity domain.



- From the menu, you select **Identity & Security** and don't see **Domains** under the **Identity** section.



- Create the required dynamic group and assign a policy to that group to allow your Oracle Integration instance to access Oracle Cloud Infrastructure Object Storage. The policy defines the permissions for the dynamic group and determines which operations the dynamic group can perform on Oracle Cloud Infrastructure Object Storage and other services such as Oracle Cloud Infrastructure Functions, Oracle Cloud Infrastructure Vision, and more
 - Log in to the Oracle Cloud Infrastructure Console.
 - Obtain the client ID of the OAuth application for the Oracle Integration instance.
 - In the upper right corner, select **Profile**, then click the identity domain.



- b. In the left navigation pane, click **Oracle Cloud Services**.



The Oracle Cloud Services page for your domain appears.

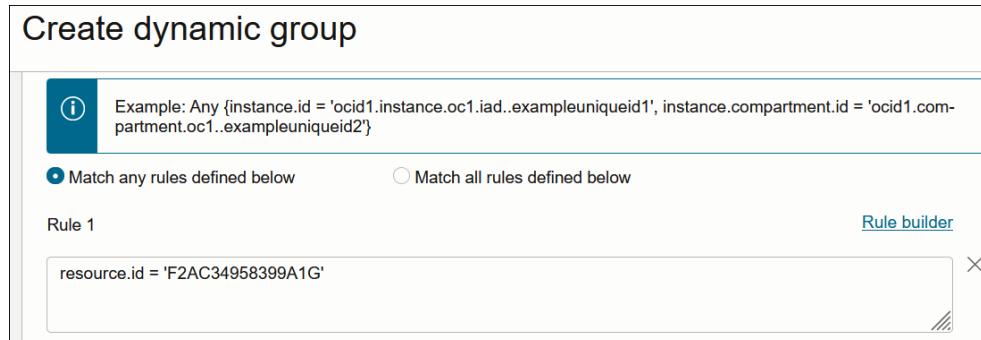
- c. In the **Name** column, click your service instance.
- d. Scroll down to the **General Information** section and copy the client ID value to use to create your dynamic group.
3. Scroll to the breadcrumbs at the top and click **Default domain**.



4. In the left navigation pane, click **Dynamic groups**.
5. Click **Create Dynamic Group**.
6. Enter the following details:
 - a. In the **Name** and **Description** fields, enter values. These fields are required.

- b. In the **Matching Rules** section, enter the required rule. The resource ID you specify must match the client ID of the OAuth application of your Oracle Integration instance. Ensure that you enclose the value in single quotes. For example:

```
resource.id = 'client_ID'
```



7. Scroll to the breadcrumbs at the top and click **Identity**.



8. In the left navigation pane, click **Policies**.
9. Click **Create Policy**.
10. Select the compartment in which to create the policy.
11. Enter the following details:
 - a. In the **Name** and **Description** fields, enter values. These fields are required.
 - b. In the **Policy Builder** section, build the required policy for the dynamic group. Examples of the minimum policy required are as follows:

```
allow dynamic-group dynamic_group to manage object-family in
compartment compartment_name
```

```
allow dynamic-group dynamic_group to inspect compartments in
compartment compartment_name
```

Where:

- *dynamic_group*: Is the dynamic group name you specified in Step 5.
- *compartment_name*: Is the compartment in which your Oracle Integration instance is located.

This enables the Oracle Integration instance associated with the dynamic group to call Oracle Cloud Infrastructure Object Storage in this particular compartment. The

RPST token is only valid for resources to which the dynamic group has been granted access using this policy.

Details about more granular permissions for object storage are provided. See [Securing Object Storage](#).

Invoke Oracle Cloud Infrastructure Object Storage from an Integration

 **Note:**

Use of an OCI Object storage action in a project is not supported.

1. Add an **OCI Object storage** action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **OCI Object storage** action to the appropriate location.
- Click  at the location where you want to add the action, then select **OCI Object storage**.

The Configure object storage panel opens.

2. Enter a name.
3. Select the resource to manage.

- **Manage buckets:** Buckets are logical containers for storing objects. Buckets are region- and compartment-specific. You can define policies at a bucket level to control access.
- **Manage objects:** Objects are stored within buckets. Objects can be any data type and consist of the object itself and any metadata.

4. If you select **Manage buckets**:

- a. From the **Operations** list, select the operation to perform.
 - **Create bucket:** Create a bucket in Oracle Cloud Infrastructure Object Storage. In the mapper, you map the bucket name to create.
 - i. Select the compartment in which to create the bucket. This selection during design-time is overridden if you map anything to the **compartment Id** field in the request mapper for this action.
 - **List buckets:** Retrieve a list of objects in a bucket in Oracle Cloud Infrastructure Object Storage. In the mapper, you map the bucket name to retrieve the objects list.
 - i. Select the compartment in which to list the buckets. This selection during design-time is override if you map anything to the **compartment Id** field in the request mapper for this action.
 - **Delete bucket:** Delete a bucket in Oracle Cloud Infrastructure Object Storage. In the mapper, you map the bucket name to delete.

- b. Click **Continue**.

5. If you select **Manage objects**:

- a. From the **Operations** list, select the operation to perform:

- **Upload object:** Upload an object into Oracle Cloud Infrastructure Object Storage. In the mapper, you map the object name to upload. This is how you update an object (it overrides an existing object with the same name). The file size limit is 1 GB. However, the API can handle objects up to 50 GB in size. See [PutObject](#).
 - **Download object:** Retrieve an object from Oracle Cloud Infrastructure Object Storage. In the mapper, you map the object name to retrieve. The size limit is 1 GB. Otherwise, the download fails.
 - **Delete object:** Delete an object in Oracle Cloud Infrastructure Object Storage. In the mapper, you map the object name to delete.
 - **List objects:** List all objects in Oracle Cloud Infrastructure Object Storage. In the mapper, you map the bucket name to retrieve the objects list.
- b. Select the compartment in which to perform the operation.
- c. Select the bucket in which to perform the operation. This selection during design-time is overridden if you map anything to the **Bucket Name** field in the request mapper for this action.
- d. Click **Continue**.
6. Review your selections on the Summary page, then click **Finish**.
This creates a map action in front of the **OCI Object storage** action.

Several use cases that natively invoke Oracle Cloud Infrastructure Object Storage are provided. See [Manage Oracle Cloud Infrastructure Buckets and Objects from an Integration](#).

Add Placeholder Notes with a Note Action

You can add placeholder notes similar to sticky notes to an integration. For example, you have not yet defined an invoke connection and want to add a placeholder note in the integration indicating that you plan to define the invoke connection later. Another integration developer reads that note and may add the invoke connection or the note reminds you to add the invoke connection at a later time when you again work on the integration.

The note action is a design-time feature that does not impact runtime. Any changes to upstream or downstream actions in your integration do not impact the note action.

1. Add a note action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Note** action to the appropriate location.
 - Click  at the location where you want to add the note action, then select **Note**.
 - Click  under **Configure Note** to enter a name and optional description for the note action.
2. Enter your notes, then click **Save**. You can add up to 256 characters.
 3. Hover over the icon to display the note text.

Translate an EDI Document with the B2B Action

You can translate a message to or from the Electronic Data Interchange (EDI) format in an integration with the B2B action.

The B2B action translates an incoming EDI document into an Oracle Integration XML message and an outgoing Oracle Integration XML message into an EDI X12 document.

1. Add a B2B action to the integration in either of the following ways:

- On the right side of the canvas, click **Actions**  and drag the **B2B** action to the appropriate location.
- Click  at the location where you want to add the for-each action, then select **B2B**.

The Configure B2B Action Wizard is displayed.

2. Enter a name and optional description.
3. Select how you want to use the B2B action, then click **Continue**.
 - **B2B Trading Partner mode:** Select to include B2B trading partner profiles and message persistence in your integration.
 - **Standalone mode:** Select to use the B2B action independently in your integration *without* a B2B trading partner profile or message persistence.
4. If you select **B2B Trading Partner mode**, follow these wizard steps:
 - a. Select the B2B message direction for this B2B action:
 - **Inbound:** The B2B message is sent from the trading partner to the host trading partner (partner where Oracle Integration is installed).
 - **Outbound:** The B2B message is sent from the host trading partner (partner where Oracle Integration is installed) to the trading partner.
 - b. Select the operation for the B2B action to perform based on the direction.

If You Selected...	Then Select an Operation...
Inbound	<ul style="list-style-type: none"> • Fetch Message: Fetches a B2B business message from the persistence store, given a message identifier. • Translate: Translates an inbound EDI message to an Oracle Integration message. • Mark As Error: Marks a B2B business message with a failed processing status.
Outbound	<ul style="list-style-type: none"> • Translate: Translates an Oracle Integration message to an outbound EDI message. • Mark As Error: Marks a B2B business message with a failed processing status. • Fetch raw messages: Fetches a B2B business message payload from persistence stored for processing.

- c. If you select either of the following, the Select Data Format page is displayed.
 - **Inbound** message direction and **Fetch Message** operation

- **Outbound** message direction and **Translate** operation
- i. Select the document definition for the B2B action to handle, and click **Search** to refresh the page.
 - ii. Select the document standard, version, and type.
 - iii. Click **>**.
- d. Review your selections on the Summary page, then click **Done**.
5. If you select **Standalone mode**, follow these wizard steps:
- a. Specify the message translation and document format details.

Element	Description
Select the direction in which to translate the message	<ul style="list-style-type: none"> • Inbound EDI message to Oracle Integration message: When an integration receives an EDI document from a business partner, it is considered an inbound document (an EDI document is translated to XML). • Oracle Integration message to outbound EDI message: When an integration sends an EDI document to a business partner, it is considered an outbound document (an EDI document is generated from XML).
Document Standard	<p>Select the document standard to follow when exchanging business documents between partners.</p> <ul style="list-style-type: none"> • Custom • Delimited • EDIFACT • FixedLength • OAG • X12 • X12HIPAA
Document Version	Select the version of the document standard to use.
Document Type	Select the document type (for example, purchase order, invoice, shipping notice, or others). The document types available for selection are based on the document version you selected.
Document Definition	<p>Select the document definition that you created on the Documents page. For EDIFACT, X12, or X12HIPAA, you can also select Standard. See Create Custom Document Definitions in <i>Using B2B for Oracle Integration 3</i>.</p>
EDI Character Encoding	Select the character encoding that the inbound EDI document is expected to use.
Perform validations on input data	<ul style="list-style-type: none"> • Yes: Validates the structure and data of an inbound EDI message. Enabling message validation has an impact on performance. If errors are found, translation does not succeed. • No: Errors are ignored during translation and the message is passed through in its current format. You can override this setting at runtime with the Validate request element in the mapper.

- b. Click **Continue**.
- c. Specify to optionally upload sample data to test that translation is successful.

Element	Description
Run a test translation with sample data (optional)	Select or drag and drop a sample file. To test inbound EDI message translation, upload an EDI document. To test outbound EDI message generation, upload an XML document.
Translate	Click to translate your sample data. Output is displayed in the Output of translation field. Any errors are displayed in the Error in translation field.

- d. Click **Continue**.
- e. Review your selections on the Summary page, then click **Finish**.

Convert HL7 Messages with a Healthcare Action

You can convert a message to or from HL7 format with a healthcare action in an integration. The healthcare action converts native inbound HL7 messages into Oracle Integration XML payloads for use in your integrations and generates native HL7 messages from Oracle Integration XML payloads to send to external applications that support HL7.

 **Note:**

If your Oracle Integration instance does not include the Healthcare edition, you cannot drag the healthcare action into an integration. See Create an Oracle Integration Instance in *Provisioning and Administering Oracle Integration 3*.

1. Add a healthcare action to the integration in either of the following ways:

- On the right side of the canvas, click **Actions**  and drag the **Healthcare** action to the appropriate location.
- Click  at the location where you want to add the healthcare action, then select **Healthcare**.

The Configure healthcare action call wizard is displayed.

2. Enter the following details:

Element	Description
Name	Enter a name.
Description	Enter an optional descriptions for what this action does.

Element	Description
Operation	<p>Select an operation to perform:</p> <ul style="list-style-type: none"> • Convert message reference to document: Use this operation to identify the incoming message type and version. This operation returns a reference to a translated message and metadata needed to route the integration to an appropriate child integration to process the message. When you call the operation, you need to specify a list of HL7 documents with which you want to match. • Match and translate inbound message: Use this operation to convert inbound HL7-formatted messages into XML format. Specify the documents you want this action to receive. These are the documents that you configured during design time. This operation takes a raw HL7 message off the wire, parses it, and matches it with the document definition you select in the Choose documents section. The message is then converted to an Oracle Integration XML payload. • Translate to outbound message: Use this operation to convert an XML payload message to an outbound HL7 message.
Choose documents This field is displayed if you select the Match and translate inbound message operation.	Select <i>all</i> the HL7 message definitions that you expect to receive on the port number you configured for your MLLP Adapter connection. This integration receives and routes all messages sent to that port number. Documents that are not selected are ignored. If you created the integration in a project, only documents available in that project are visible for selection.
Document This field is displayed if you select the Convert message reference to document or Translate to outbound message operation.	Select the document to use. If you created the integration in a project, only documents available in that project are visible for selection.

For example:

- If you select **Match and translate inbound message**, the selected ADT_A08 message document is converted from HL7 format to an Oracle Integration XML-formatted message.

The screenshot shows the 'Configure' screen for a 'Configure healthcare action call'. The interface includes fields for Name ('patientUpdates'), Description ('Update patient details'), and Operation ('Match and translate inbound message'). Below these, there's a 'Choose documents' section with a search bar ('Search available documents') and a document selection grid. The 'Available' column contains 'VLANTIGU_DOC'. The 'Selected' column contains 'MYADT_A08_DOC'. Between the two columns are four buttons: a right-pointing arrow, a double-right arrow, a left-pointing arrow, and a double-left arrow.

Name
patientUpdates

Description
Update patient details

Operation
Match and translate inbound message

Choose documents

Search available documents

Available	Selected
VLANTIGU_DOC	MYADT_A08_DOC

> >> < <<

- If you select **Translate to outbound message**, the selected ADT_A08 message is converted from an Oracle Integration XML message to an HL7-formatted message for delivery to an external application that supports HL7.

Configure
Configure healthcare action call

Name: patientUpdates

Description: Update patient details

Operation: Translate to outbound message

Document: MyADT_AO8_Doc

Selected Document Definition has Standard 'HL7V2', Version '2.9', Type 'ADT_A08 (Update patient information)'.

- If you select **Convert message reference to document**, the message reference is converted into a mappable payload. This operation returns the contents of the HL7 message as a variable in your integration that is ready to use in the business logic and mapper of your integration. The variable appears on the source side of the mapper.

Configure
Configure healthcare action call

Name: patientupdates

Description:

Operation: Convert message reference to document

Document: MyADT_AO8_Doc

Selected Document Definition has Standard 'HL7V2', Version '2.9', Type 'ADT_A08 (Update patient information)'.

- Click **Finish**.

Examples of using all three operations in a use case are provided. See Design Time and Runtime Overview of an Oracle Integration for Healthcare Use Case.

Interact with Files in File Server

You can obtain a list of files, write or move a file to File Server, get the reference of a file and then stage read or unzip the file, and delete a file on File Server using the File server action.

On This Page

- [What You Can Do with the File Server Action](#)
- [When Not to Use the File Server Action](#)
- [Start Here: Add the File Server Action to an Integration](#)
- [Option 1. Obtain a List of Files in File Server](#)
- [Option 2. Write a File to File Server](#)
- [Option 3. Move a File to File Server](#)
- [Option 4. Stage a File and Work with It, or Write a File to Another Location in File Server](#)
- [Option 5. Delete a File in File Server](#)
- [Next Steps](#)

What You Can Do with the File Server Action

The File server action lets you interact with files in the following ways.

For all of the options, you can hard code values, but most of the time, you'll pass dynamic values into the File server action using the Map action. Complete that task after you finish defining the File server action. See [Next Steps](#).

Goal	Operation to use	Step-by-step instructions
Obtain a list of files in a File Server directory	List directory	<ol style="list-style-type: none">1. Start Here: Add the File Server Action to an Integration2. Option 1. Obtain a List of Files in File Server
Create a new file in a File Server directory	Write File	<ol style="list-style-type: none">1. Start Here: Add the File Server Action to an Integration2. Option 2. Write a File to File Server
Move a file from one File Server directory to another (similar to a cut and paste operation)	Move File	<ol style="list-style-type: none">1. Start Here: Add the File Server Action to an Integration2. Option 3. Move a File to File Server

Goal	Operation to use	Step-by-step instructions
Get a reference to a file in a File Server director, and use the reference to complete either of the following tasks: <ul style="list-style-type: none">• Stage the file and then work with the staged file as needed in Oracle Integration• Write the file to another File Server directory	Get File Reference	<ol style="list-style-type: none"> 1. Start Here: Add the File Server Action to an Integration 2. Option 4. Stage a File and Work with It, or Write a File to Another Location in File Server
Encrypt or decrypt a file	<ul style="list-style-type: none"> • For files that must be encrypted and saved to File Server: Get File Reference, which gets a reference to a file. After the File server action, add a stage file action to encrypt the file, and then another File server action using the Write File operation. The File server action uploads the file to File Server by passing the encrypted file reference. • For encrypted files in File Server that must be decrypted: Get File Reference, which gets a reference to an encrypted file, followed by a stage file action to decrypt the file. <p>You can also use the capabilities of the FTP Adapter to encrypt or decrypt a file.</p>	<ol style="list-style-type: none"> 1. Start Here: Add the File Server Action to an Integration 2. Option 4. Stage a File and Work with It, or Write a File to Another Location in File Server 3. Process Files in Schedule Integrations with a Stage File Action 4. Option 2. Write a File to File Server <p>Alternatively, to use the FTP Adapter to encrypt or decrypt a file, see <i>Create a Connection and Add the FTP Adapter Connection to an Integration</i> in <i>Using the FTP Adapter with Oracle Integration 3</i>.</p>
Delete a file in a File Server directory	Delete File	<ol style="list-style-type: none"> 1. Start Here: Add the File Server Action to an Integration 2. Option 5. Delete a File in File Server

When Not to Use the File Server Action

If you need to complete the following tasks, use a connection that is based on the FTP Adapter, or use the stage file action:

- Write a file to a temporary storage location and then perform work on the file, such as by updating the file.
- Sign or verify a file.

Start Here: Add the File Server Action to an Integration

No matter what you need to use the File server action to do, start here by adding the action to your integration.

1. Add a File server action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **File server** action to the appropriate location.
- Click  at the location where you want to add the assign action, then select **File server**.

 **Note:**

If you don't see the File server action or can't add it to an integration, then an administrator needs to enable File Server for your Oracle Integration instance. See [Enable File Server in Using File Server in Oracle Integration 3](#).

2. Fill in the following fields.

Field	Description
What do you want to call your endpoint?	Name of the endpoint, such as <code>write_file_to_File_Server</code> .
Select resource	Select one of the following options: <ul style="list-style-type: none"> • File: Interact with a file, such as by writing, moving, downloading, reading, or deleting it. • Directory: Obtain a list of files in a specific directory in File Server.

Additional fields, which are related to the operation that you selected, appear.

3. Continue following the steps in the appropriate section:

- [Option 1. Obtain a List of Files in File Server](#)
- [Option 2. Write a File to File Server](#)
- [Option 3. Move a File to File Server](#)
- [Option 4. Stage a File and Work with It, or Write a File to Another Location in File Server](#)
- [Option 5. Delete a File in File Server](#)

Option 1. Obtain a List of Files in File Server

1. Complete the steps in [Start Here: Add the File Server Action to an Integration](#).
2. From the **Select operation** drop-down list, select **List directory**.
3. Fill in the following fields.

Field	Description	Can override later in the mapper?
Input Directory	Name of the directory that contains the files that you need to list. Oracle recommends using an absolute directory. For example, <code>/HOME/USERS/</code> .	Yes
File name pattern	Pattern of the files that you want to list, such as <code>*.csv</code> for only CSV files. Or, enter <code>*</code> to list files of all types.	Yes

Field	Description	Can override later in the mapper?
Max Files	<p>Maximum number of files to return, up to 1000.</p> <p>Note: The File server action returns the files according to the last modified time. If you selected 10 as the maximum number of files and the last modified time of the eleventh file is the same as the tenth file, then the action also returns the eleventh file and any other files with the same last modified time. For example, consider a directory with 15 files, and the Max Files setting is 10. If the tenth, eleventh, twelfth, and thirteenth files have the same time stamp, then the File server action returns thirteen files.</p>	No
Minimum Age (seconds)	<p>Minimum age in seconds of the files to return.</p> <p>For example, if a file was last modified at 02:28:45 AM, and the minimum age is 80 seconds, the file isn't returned at 02:29:00 AM and 02:30:00 AM. The file is returned only after 02:30:05.</p>	No
List File Recursively	Whether to list files within directories recursively beneath the input directory.	No

4. Select **Continue**.
5. Review the summary of the action, and select **Finish**.

The File server action appears on the canvas with a Map action before it.

Option 2. Write a File to File Server

1. Complete the steps in [Start Here: Add the File Server Action to an Integration](#).
2. From the **Select operation** drop-down list, select **Write File**.
3. Fill in the following fields.

Field	Description	Can override later in the mapper?
Specify an Output Directory	Name of the directory to write the files to. Oracle recommends using an absolute directory. For example, /HOME/USERS/.	Yes
File Name	Name of the file to create.	Yes

4. Select **Continue**.
5. Review the summary of the action, and select **Finish**.

The File server action appears on the canvas with a Map action before it.

Option 3. Move a File to File Server

1. Complete the steps in [Start Here: Add the File Server Action to an Integration](#).
2. From the **Select operation** drop-down list, select **Move File**.
3. Fill in the following fields.

Field	Description	Can override later in the mapper?
Directory path	Name of the directory that contains the file to be moved. Oracle recommends using an absolute directory. For example, /HOME/USERS/.	Yes
File name	Name of the file to be moved. The file can be of any size. If the file is missing, an error occurs when the integration attempts to locate it.	Yes
Target directory path	Name of the directory to move the file to. Oracle recommends using an absolute directory. For example, /HOME/USERS/. If the directory doesn't exist, File Server creates it.	Yes
Target file name	Name of the file in the target directory.	Yes
Overwrite File	When selected, the target file is overwritten if it already exists. If you deselect this option and the file already exists in the target directory, an error occurs when File Server attempts to move the file to the directory.	No

4. Select **Continue**.
5. Review the summary of the action, and select **Finish**.

The File server action appears on the canvas with a Map action before it.

Option 4. Stage a File and Work with It, or Write a File to Another Location in File Server

This operation gets a reference to a file in File Server. If anyone moves or deletes the file while the integration is running, an error occurs, indicating that the file doesn't exist.

1. Complete the steps in [Start Here: Add the File Server Action to an Integration](#).
2. From the **Select operation** drop-down list, select **Get File Reference**.
3. Fill in the following fields.

Field	Description	Can override later in the mapper?
Input Directory	Name of the directory that contains the files that you need to download. Oracle recommends using an absolute directory. For example, /HOME/USERS/.	Yes
File Name	Name of the file in the input directory.	Yes

4. Select **Continue**.
5. Review the summary of the action, and select **Finish**.

The File server action appears on the canvas with a Map action before it.

Option 5. Delete a File in File Server

1. Complete the steps in [Start Here: Add the File Server Action to an Integration](#).
2. From the **Select operation** drop-down list, select **Delete File**.

3. Fill in the following fields:

Field	Description	Can override later in the mapper?
Directory path	Name of the directory to delete. Oracle recommends using an absolute directory. For example, /HOME/USERS/.	Yes
File name	Name of the file to delete.	Yes

4. Select **Continue**.

5. Review the summary of the action, and select **Finish**.

The File server action appears on the canvas with a Map action before it.

Next Steps

- Pass dynamic values into the File server action using the Map action, which appears before the File server action on the canvas. See About Mapping Data Between Applications in *Using the Oracle Mapper with Oracle Integration 3*.
 - If you need to encrypt a file and then write it to File Server, perform the following steps:
 1. After the File server action, add a stage file action to encrypt the file.
See [Process Files in Schedule Integrations with a Stage File Action](#).
 2. After the stage file action, add another File server action using the Write File operation. This action uploads the file to File Server by passing the encrypted file reference.
See [Option 2. Write a File to File Server](#).
- Alternatively, use the FTP Adapter to encrypt the file and write it to File Server. See Create a Connection and Add the FTP Adapter Connection to an Integration in *Using the FTP Adapter with Oracle Integration 3*.
- If you need to decrypt a file that is in File Server, use the stage file action or the FTP Adapter. See the following links:
 - [Process Files in Schedule Integrations with a Stage File Action](#)
 - Create a Connection and Add the FTP Adapter Connection to an Integration in *Using the FTP Adapter with Oracle Integration 3*

Call a Robot from an Integration

A robot runs only when an integration calls it. Call a robot from an integration using the robot process automation action.

All information about robots is in a separate guide. For guidance about using the robot process automation action, see Design an Integration That Calls a Robot in *Using Robots in Oracle Integration 3*.

Use XPath Axis and Wildcard Expressions in the Expression Builder

You can include XPath axis and wildcard expressions in actions that support the Expression Builder (for example, assign and switch actions support the Expression Builder).

1. Create an assign action or switch action and navigate to the Expression Builder.
2. For an assign or switch action, build an expression using either option:
 - Wildcard expression:

For this example, a wildcard is entered to select all elements:

Answer

```
/nssrcmpr:process/nssrcmpr:Answer/*
```

Or, to select all elements:

```
/*
```

- Axis expression:

For this example,

descendant

is entered to select all descendants (child, grandchildren, and so on) of the current node. Any descendant with this ID and namespace:

```
mb_v1_3:ID
```

is retrieved by the expression.

```
/nssrcmpr:process/nssrcmpr:Answer/descendant::mb_v1_3:ID
```

Axis expressions adhere to the following syntax:

```
axisname::nodetest[predicate]
```

The following table provides examples of axis expressions:

Syntax	Result
child::book	Selects all book nodes that are children of the current node.
attribute::lang	Selects the lang attribute of the current node.

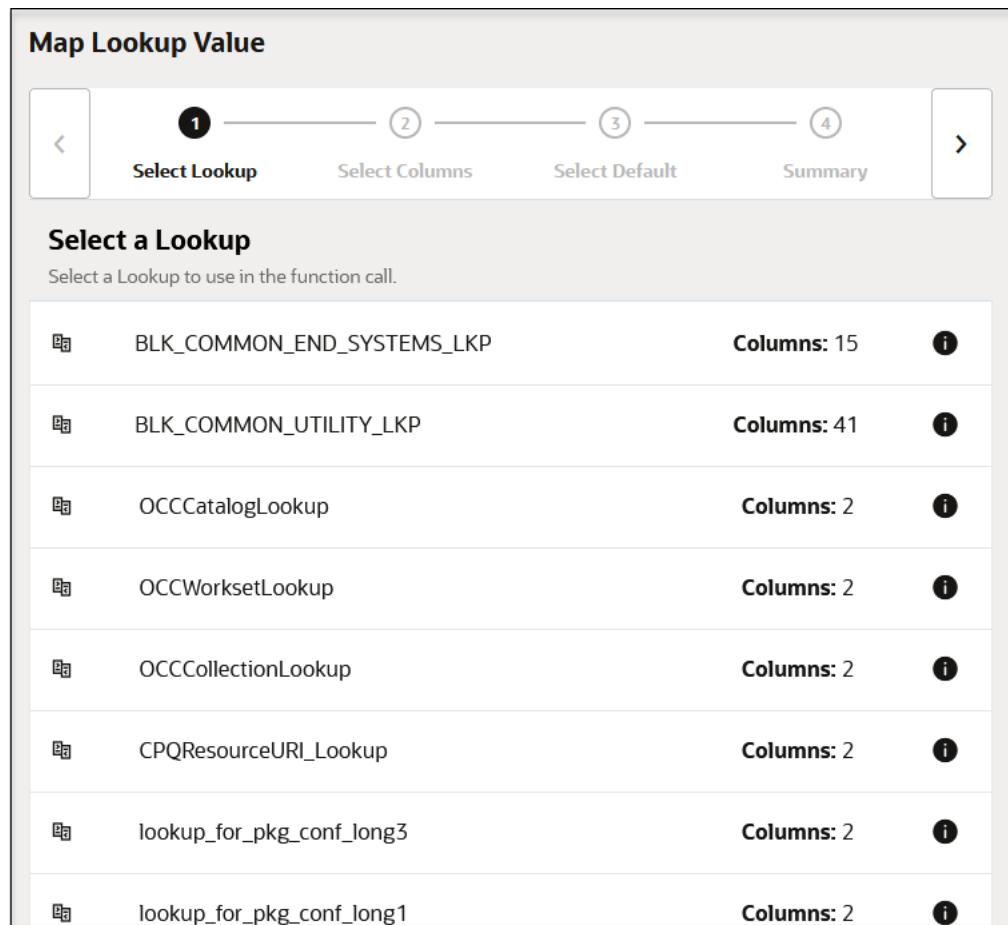
Syntax	Result
child::>*	Selects all element children of the current node.
attribute::*	Selects all attributes of the current node.
child::text()	Selects all <code>text</code> node children of the current node.
child::node()	Selects all children of the current node.
descendant::book	Selects all <code>book</code> descendants of the current node.
ancestor::book	Selects all <code>book</code> ancestors of the current node.
ancestor-or-self::book	Selects all <code>book</code> ancestors of the current node and the current if it is a <code>book</code> node.
child::*/child::price	Selects all <code>price</code> grandchildren of the current node.

Use Lookups in Variable Assignments

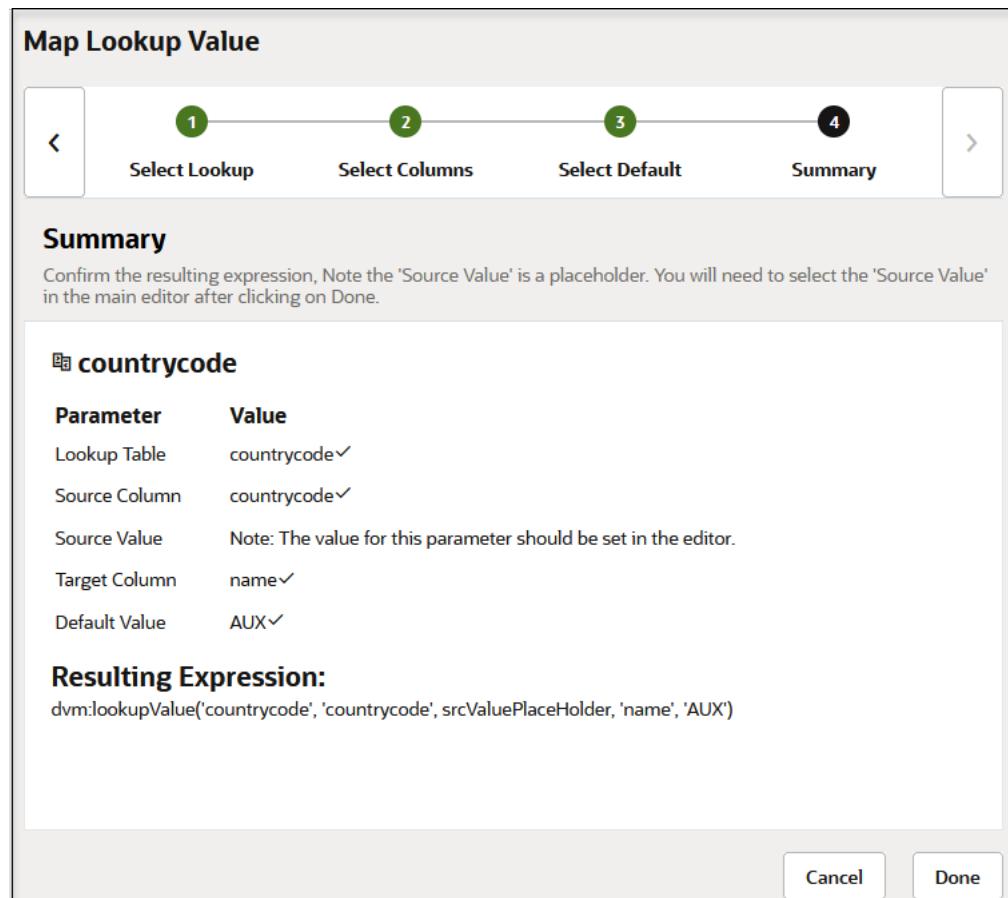
You can create variable assignments that use lookups in the Expression Builder. You must have already created the lookup.

1. Create a lookup in Oracle Integration. See [Manage Lookups](#).
2. Drag an **Assign** action to the + sign. You can also create variable assignments that use lookups in other actions that use functions (for example, a data stitch action).
3. Enter a name and optional description.
4. Select **String** from the **Datatypes** drop-down list.
5. Click **Switch to Developer Mode** 
6. Click the **Functions** tab in the **Sources** tree.
7. Drag the **lookupValue** function to the **Value** field. You can use search to access this function.

This starts the Map Lookup Value Wizard.



8. Select the lookup to use, and click **Next >**.
9. Select the source and target columns, and click **Next >**.
10. Enter a default value, and click **Next >**.
11. Review your selections on the Summary page and note the **srcValuePlaceHolder** value to replace in the expression, then click **>**.



12. Click the **Input Sources** tab.
13. In the **Sources** tree, drag a value (for this example, **\$tracking_var_1**) on top of the **srcValuePlaceHolder** value.

The expression appears as follows:

```
(x) Value
$tracking_var_1$dvm:lookupValue('countrycode',
'countrycode', $tracking_var_1, 'name',
'AUX')
```

14. Click **Save.**

Lookup functions are available for use in the target tree of the mapper.

Define Fault Aggregation in Parallel Processing Scenarios

Oracle Integration includes a predefined fault object for fault aggregation. This object enables you to catch and aggregate faults in the context of parallel processing in schedule integrations and to send these faults to an external service to define more granular error handling. To define a fault object and aggregate the faults, you must use actions that support looping (for example, for-each loops, while loops, and so on).

Note:

The fault object is only supported with schedule integrations.

Two fault aggregation operations are available for selection in schedule integrations:

- **append:** Adds a new fault to the end of a fault list.

- **clear**: Removes all entries in a fault list.

These options are available for selection under the following scenarios:

Location	Append Option Allowed?	Clear Option Allowed?
Top level (that is, outside of any looping actions)	Yes	Yes
Inside a for-each action with the Process items in parallel option selected	Yes	No
Inside a for-each action loop and a while action loop	Yes	Yes
Inside a stage file action configured with the Read File in Segments operation	Yes	No
Inside a scope action	Yes	Yes
Inside a global fault or a named fault (that is, inside a scope fault handler)	Yes	Yes

1. Create a schedule integration.
2. Create an aggregated fault per the scenarios supported in the above table. For example, create a for-each action, then drag an assign action within the loop.
3. Create an assignment and select **faults** from the **Datatypes** list.

The page is refreshed to display the **Operations** list with two options.



4. Build the fault object mapping.
5. Complete your mapping.
6. Define an invoke connection to send the aggregated faults to an external service.

 **Note:**

Changing the name, data type, or both is only allowed during initial creation of the variable while not yet assigned a value.

Assign 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:

- [Assign Business Identifiers](#)
- [Delete Business Identifiers](#)
- [Maintain Business Identifier Definitions During Trigger Connection/Schedule Integration Parameter Changes](#)

Assign 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 Instances page and (if integration errors have occurred) the Errors page. If you created scheduled parameters, they are available for assignment as business identifiers.

If you have not yet configured at least one business identifier tracking field in your integration, an error icon is displayed in the design canvas. See [Missing a Primary Business Identifier in an Integration](#).

1. In the navigation pane, click **Design**, then **Integrations**.
2. 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.

3. Click **Business Identifiers** .

The Business Identifiers panel 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.

4. From the **Sources** tree, drag the payload field that you want to track to **Business Identifier Field**. At least one identifier is required. If you only add one, it is automatically selected as the primary key. A check mark to the left of the name indicates that this is the primary business identifier. If you created scheduled parameters, they are available for assignment as a primary or secondary business identifier.

5. Drag additional business identifiers to the fields, when needed.

Delete Business Identifiers

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

1. In the navigation pane, click **Design**, then **Integrations**.
2. Click the specific integration in which to delete a business identifier. You can only delete business identifiers in integrations that are *not* active. If an integration is active, you can view, but not delete, the contents of the Business Identifiers panel.

3. Click **Business Identifiers**

The Business Identifiers panel is displayed.

4. For the business identifier to delete, click **Clear** . 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 Instances page.

Maintain Business Identifier Definitions During Trigger Connection/Schedule Integration Parameter Changes

Beginning with new integrations created in Release 23.08, when a trigger connection changes or a schedule integration parameter is deleted, business identifier definitions are not removed. Instead, the definitions are retained and validated based on the changes.

The following table describes business identifier definition status based on the release in which the integration was created.

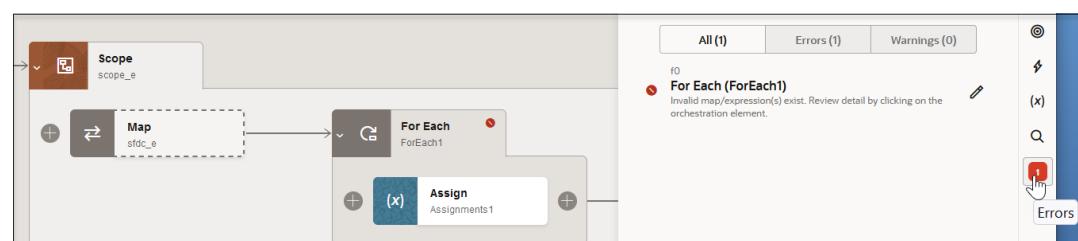
If You ...	Existing Integrations Created Prior to Release 23.08	New Integrations Created in Release 23.08 and Later
Edit a trigger connection in the Adapter Endpoint Configuration Wizard and either make changes or simply click through the wizard pages without making changes.	The business identifier is removed. You must redefine the business identifier. After that, the business identifier behavior described for new integrations created in Release 23.08 is followed.	The business identifier is retained and validated. If you remove a parameter used by the business identifier (for example, a query parameter in the wizard that is used as the business identifier definition), a validation error is displayed in the Errors panel:
		<p>Primary business identifier Invalid tracking variables definition</p> <p>Re-define the business identifier to resolve the error.</p>
Delete a scheduling parameter used by the business identifier.	Validation is performed and an error is displayed in the Errors panel.	Validation is performed and an error is displayed in the Errors panel.

Display Errors and Warnings in an Integration

If there are errors or warnings in an integration (for example, an empty or invalid map, a missing tracking attribute, or an invalid assign or switch action), an **ERRORS** section is displayed on the side. These errors and warnings prevent you from activating an integration. You must first resolve these issues to activate an integration.

1. Design an integration.

If there are errors or warnings, an **Errors** section is displayed on the side.



2. Click the **Errors** section to display error and warning details. For this example, a for-each action has an invalid mapping/expression.

3. Click **Edit** to access the page to edit.

4. Resolve the issue. Once successfully resolved, the **ERRORS** section disappears.

You can now activate the integration.

Create Schedule Integrations

You can schedule the running of integrations. For example, you can create an integration that is triggered by a schedule in which an FTP Adapter reads a file and another FTP Adapter writes the file. You can schedule this integration run to copy files at a date and time of your choosing. You can also define the frequency of the integration and run ad-hoc, schedule integrations. When you create a schedule integration, a schedule icon is displayed with the integration entry on the Integrations page.

Topics:

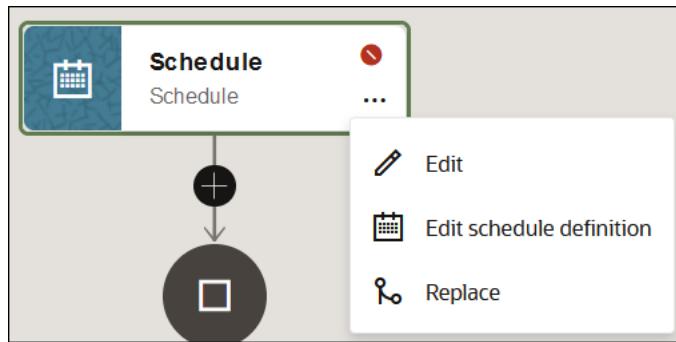
- [Create a Schedule Integration](#)
- [Define the Integration Schedule](#)
- [Create and Update Parameters in Schedule Integrations](#)
- [Run a Schedule Integration Immediately](#)
- [Start and Pause an Integration Schedule](#)
- [Change the User Submitting an Integration Schedule](#)
- [Globally Change the Submitter of Integration Schedules](#)
- [View Past and Present Schedule Integration Runs](#)
- [View Future Schedule Integration Runs](#)
- [Edit an Integration Schedule](#)
- [Delete an Integration Schedule](#)
- [Unlock an Integration Schedule](#)
- [Convert a Schedule Integration to a REST Adapter-Triggered, Application Integration](#)
- [Delete Scheduled or Ad-Hoc Runs](#)
- [Monitor Schedule Integration Runs](#)
- [Understand Schedule Integration Behavior When Using the Local Time Zone](#)

See *Understand the FTP Adapter* in *Using the FTP Adapter with Oracle Integration 3*.

Create a Schedule Integration

This section describes how to create a schedule integration.

1. Follow the steps in [Create an Integration](#) to create a schedule integration. An empty integration canvas is displayed with a schedule icon at the top.
2. To define the schedule for running the integration, click **Actions ...**, then select **Edit schedule definition**.



See [Define the Integration Schedule](#). You can also define a schedule on an integration at a later time from the Integrations page by clicking **Actions ...**, then selecting **Add schedule**.

3. See [Design an Application Integration](#) to design the entire flow of your schedule integration and [Assign Business Identifiers for Tracking Fields in Messages](#) to assign business identifiers. Schedule integration design differs from application integration design only in that you define an adapter trigger connection in an application integration. In schedule integrations, the schedule is the trigger connection in the integration.

Define the Integration Schedule

You can define a schedule for running integrations, such as the frequency at which to run the integration, whether to run the schedule based on an iCal expression, whether the schedule run never expires, whether the schedule has a fixed expiration date, and so on.

Note:

- You cannot separately export the schedule of an integration.
- When the user that created and scheduled an integration is locked or deleted, the schedule does not run. Ensure that this user is not locked or deleted.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Find the schedule integration in the Integrations list.

These integrations are identified by **Schedule integration** . There are several ways in which to create a schedule on an integration.

3. If you want to first design and activate the integration and then create the schedule:
 - a. Hover over the integration.
 - b. Click **Activate** .

The Activate Integration dialog is displayed.

 - c. Click **Activate And Schedule**.
4. If you want to first create a schedule for an integration that you design and activate later:
 - a. Hover over the integration on which to create a schedule.

- b. Click **Actions ...**, then select **Add Schedule**.

 **Note:**

For integrations with no defined schedules, the menu shows **Add Schedule**.
For integrations that have a defined schedule, the menu shows **Schedule**.

The page for defining the schedule processing details is displayed.

 **Note:**

- Assume you create a schedule and generate several integration runs, then deactivate and delete the integration. If you then create and activate a new integration with the same name as the deleted integration, and go to the Instances page, the past runs of the deleted integration are displayed. This is by design.
- Deactivation of an active, schedule integration is not allowed. You must stop the schedule, then retry.

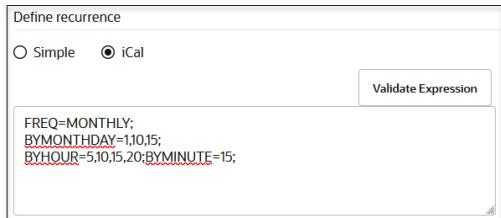
5. If you want to schedule basic integration runs, click **Simple**. For this type, there is a minimum frequency limit of ten minutes. If you try to define a schedule frequency of under ten minutes, a validation error occurs.
- a. In the **Frequency** section for a **Simple** schedule, click the list to select the frequency with which to run the integration. When you define one frequency, you can specify additional values by clicking the icon to the right of the **Frequency** section.
 - b. Select a frequency.
 - **Only Once**: This is the default selection.
If you select this option, you cannot select **When schedule starts** as the start date. This option is disabled.
 - **Hours and Minutes**: Specify the hour and minute 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.
6. If you want to schedule integration runs with an iCal expression, click **iCal**.

 **Note:**

- Oracle Integration allows up to five recurrences in an iCal expression when defining the schedule.
- There is a one minute limitation on how frequently you can run schedule integrations with an iCal expression. Anything under this limit is not supported.
- You cannot use lookups in iCal expressions.

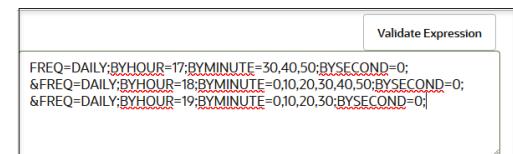
- a. Enter an iCal expression, and click **Validate Expression**. For example:

Expression	Syntax
This expression indicates that this integration runs each month on the 1st, 10th, and 15th days of the month at 5:15 AM, 10:15 AM, 3:15 PM, and 8:15 PM.	FREQ=MONTHLY; BYMONTHDAY=1,10,15; BYHOUR=5,10,15,20;BYMINUTE=15;



This expression defines multiple schedule frequencies. The schedule runs every day between the hours of 5:30 PM – 7:30 PM, and during these hours it processes every 10 minutes. This configuration requires three schedules separated by the & sign:

```
FREQ=DAILY;BYHOUR=17;BYMINUTE=30,40,50;BYSECOND=0;
&FREQ=DAILY;BYHOUR=18;BYMINUTE=0,10,20,30,40,50;BYSECOND=0;
&FREQ=DAILY;BYHOUR=19;BYMINUTE=0,10,20,30;BYSECOND=0;
```



This expression runs daily at 8 AM and also monthly at 12 PM on day 1 and day 2.

```
FREQ=DAILY;BYHOUR=8;&FREQ=MONTHLY;
BYMONTHDAY=1,2;BYHOUR=12;
```



- b. If validation is successful, the following message is displayed at the top: iCal expression is valid.
7. In the **This schedule is effective** section, click the **When schedule starts** link to the right of **From**.
- A menu is displayed for defining the start date of the schedule.
8. If you want to start the integration run when the schedule is activated, click **When schedule starts**.
9. 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 **Done**.
10. In the **Until** section, click the **Never (repeat indefinitely)** link to the right.
A menu is displayed for defining the expiration date.
11. If you want the schedule run to never expire, select **Never (repeat indefinitely)**.
12. 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 **Done**.
13. From the **Time zone** list, select a time zone in which to run the schedule. By default, this field shows the value you selected on the Preferences page.
14. Click **Save**.

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

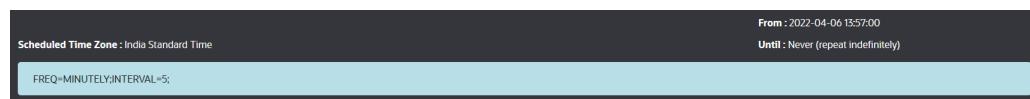
Schedule Run name saved successfully

If there are any errors, a validation message is displayed in the upper right corner that describes how to resolve the errors.

15. Return to the Schedule and future runs page.

Details about the integration run schedule are displayed. There are two sections:

- Schedule section: This section is collapsed by default. Clicking the top bar (showing schedule name and buttons) expands the section and shows the schedule.



- Future Instances table: This table remains empty until the schedule is started. Once it starts, you see data.

288 Future Instances		
Type	State	Scheduled Time
Scheduled Run	Yet To Start	in 4 minutes
Scheduled Run	Yet To Start	in 9 minutes
Scheduled Run	Yet To Start	in 14 minutes

16. If you have already activated the integration, select an option to run the integration:

- a. Click **Actions** , then select **Run**.

The Configure and run page is displayed.
Clicking **Run** triggers the integration immediately.

- b. If you are a user with the ServiceAdministrator role, you can change the user with which to run the schedule in the **Run as** field.
 - c. Select how to initiate an instance of the integration when prompted.
 - **Ad hoc request:** The instance is processed independently of the scheduled run for the integration (if any).
 - **As part of schedule:** The instance runs in sequence with the scheduled run and shares schedule parameters (if any).
 - d. Click **Run**.
- or
- a. Select **Start Schedule**  on the Schedule and future runs page to activate the integration schedule.
 - b. If you are a user with the ServiceAdministrator role, you can change the user with which to submit the schedule.
 - c. Click **Start Schedule** when prompted. Otherwise, click **Cancel**.

Note:

- If you place a file in an input directory and run an integration for the first time, the file is copied to the output directory. If you then run the integration a second time, the same file is not copied again to the output directory, even if you deleted the file from the output directory before rerunning the integration. This is by design. Second integration runs do not copy the same file. However, if a new file is placed in the input folder or the previously-copied file is updated with a newer version in the input directory, both are copied to the output directory.
- If you create a new schedule to repeat every minute and launch the calendar to specify a start time, it shows the current time (for example, 12:41:16). If you update the start time to several minutes later (for example, 12:43:55), save, and start the schedule integration run, the integration starts running at 12:43:16 instead of 12:43:55. The scheduler does not care about the seconds entered and runs the schedule every minute.

17. If you have not yet activated the integration, return to the Integrations page and click  **Activate**.
18. Run the schedule integration as described in the previous steps.

Create and Update Parameters in Schedule Integrations

You can create and update scalar-type scheduled parameters in schedule integrations that determine how to batch and read data received from a source location. You then use these parameter values downstream in the integration.

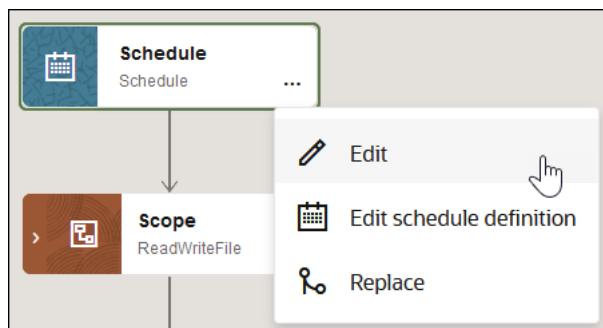
You create and assign values to these parameters in the Schedule parameters page that is available in schedule integrations. You can also assign values to these parameters in assign actions. Up to five parameters are supported in the Configure schedule parameters panel.

Create Parameters

You can also create and assign scheduled parameters as business identifiers that enable you to track fields in messages during runtime. See [Assign Business Identifiers](#).

Scheduled parameters are available across all scheduled runs of an integration and can facilitate processing of data from one run to the next. For example, when performing batch processing, a scheduled parameter can track the current position of batch data between runs. This value is available across scheduled runs and cannot exceed 256 characters.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Click **Create**.
3. Select the **Schedule** pattern.
4. Complete the fields of the dialog, and click **Create**.
The integration canvas is displayed.
5. For **Schedule**, click ..., then select **Edit**.



6. Click the field to open fields for entering a parameter name, description, and default value.
7. Enter values. For this example, a parameter of ATOMLastRunDateTime and a value of "2018-01-01T00:00:00.000Z" are entered.

Configure Schedule Parameters

Scheduled parameters are available across all scheduled runs of an integration and can be used to facilitate processing of data from one run to the next. For example, when performing batch processing a schedule parameter can be used to track the current position of batched data between runs.

ATOMLastRunDateTime	<input type="button" value="Delete"/>
Parameter Name *	<input type="text" value="ATOMLastRunDateTime"/>
Description	<input type="text" value="Last successful processed ATOM"/>
Default Value *	<input 2018-01-01t00:00:00.000z\""="" type="text" value="\"/>

8. Click **Save**.

If you enter a parameter without a value, a red warning icon is displayed after the **Default Value** field.

9. Drag an **Assign** action into the integration.
10. Drag the parameter you created in the Configure schedule parameters panel into the **Variable** field and specify a value.

The screenshot shows the Oracle Integration Designer interface. On the left, there's a sidebar with 'Input Sources' and 'Functions' tabs. Under 'Sources', several options are listed: 'schedule *', 'startTime', '\$getNewHireATOMFeed', '\$getNewHireATOMFeed_REQUEST', '\$getTimestamp', '\$self', '\$ATOMLastRunDateTime', '\$CurrentPotentialError', '\$newHiresCount', '\$tracking_var_1', '\$tracking_var_2', and '\$tracking_var_3'. The main area is titled ': Configure Assignment' and contains a section for 'assignATOMLRD'. It includes fields for 'Enter Description', 'Specify as many variables for this assignment as you need, and configure their values.', and a configuration panel with variables and operations. The configuration panel shows 'Variable ATOMLastRunDateTime' assigned to 'Value output_getTimestamp_dt' via an 'Assign' operation.

11. Click **Save**.

Note:

If you create a for-each action without selecting the **Process items in parallel** option and then add an assign action within the for-each loop, parameters defined in schedule integrations are visible for selection in the assign action. If you later edit the for-each action and select the **Process items in parallel** option, a red fault icon is displayed on the impacted assign action indicating that a scheduled parameter is being used. You must correct this error for your integration to be valid. You can disable the **Process items in parallel** option or use the cut and paste options to move the assign action outside of the for-each loop.

Update Parameters

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the integration on which the scheduled run is defined.
3. Click **Actions ...**, then select **Schedule**.
The Schedule and future runs page is displayed.
4. Click **Actions** and select **Update schedule parameters**.

5. Edit the parameter value in the **New value** column, and click **Update**. The value in the **New value** field replaces the currently stored value and is used for the next run. The value in the **Current value** field indicates the currently stored value.

Configure values for Schedule Parameters. New Value replaces the currently stored value and will be used for the next run. Current Value indicates the currently stored value.			
1 Services			
Parameter Name	Default Value	Current Value	New Value
ATOMLastRunDateTime	''2018-01-01T00	''2018-01-01T00	<input type="text" value="Enter new value"/>

Override Schedule Parameter Values

You can override schedule parameter values set at design-time when submitting an ad-hoc schedule request or starting a schedule in an activated integration. This feature enables you to provide parameter values while invoking an integration without deactivating it.

1. Access the dialog to override parameter values in either of two ways. If the schedule integration has defined parameters, the Configure schedule parameters panel is displayed. Otherwise, the open is not displayed and the schedule integration starts.
 - a. If submitting an ad-hoc schedule integration, select **Run** from one of the following locations:
 - Click **Actions** , then **Run** on the Schedule and future runs page.
 - Click **Actions** , then **Run** on the Monitor integrations page.
 Both options invoke the Configure and run page, from which you can select **Ad hoc request** and click **Run**.
 - b. If scheduling an integration, select **Start schedule**  on the Schedule and future runs page.

The Schedule parameters page is displayed.

Schedule Parameters - Test Oracle HCM FTP Directory Synchronization (1.0.1)			
Configure values for Schedule Parameters. New Value replaces the currently stored value and will be used for the next run. Current Value indicates the currently stored value.			
1 Services			
Parameter Name	Default Value	Current Value	New Value
ATOMLastRunDateTime	''2018-01-01T00	''2018-01-01T00	<input type="text" value="Enter new value"/>

2. Make updates to any necessary parameters.

The following fields are displayed:

- **Parameter name**: Displays the parameter name you provided during design-time.
- **Default value**: Displays the parameter value you assigned during design-time.
- **Current value**: Displays the value most recently used at runtime.
- **New value**: Optionally enter new values for the schedule parameters. These values override the current and default values. If these fields remain empty, the current values are used. If there are no new or current values, the default values defined during design-time are used.

3. In the **New value** row for the specific parameter, enter a value.

If the integration updates these schedule parameter values using an assign action, the updated values are saved and become the current values for the next schedule run.

4. Save your updates.
 - a. If starting a schedule, click **Start schedule**.
 - b. If submitting an ad-hoc schedule integration, click **Run**.

The values you entered are used during processing of the schedule integration.

Run a Schedule Integration Immediately

You can run a schedule integration immediately, known as an ad-hoc request.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the row of the schedule integration to run.
3. Click **Actions ...**, then select **Run**.

The Configure and run page is displayed.

4. If you are a user with the ServiceAdministrator role, you can change the user with which to submit the schedule.
5. Ensure **Ad hoc request** is selected. This is the default selection.
6. If any schedule parameters are defined for this integration, they are displayed after. Accept the default value or enter a different value to use for this run in the **Current value** field. Otherwise, this section is empty.
7. Click **Run**.

The Activity Stream panel opens on the right to show the progress of the run.

The screenshot shows the Oracle Workday Integration interface. On the left, the 'Configure and run' page is displayed with fields for 'Run as' (User: icsadmin), 'Request type' (Ad hoc request), and 'schedule_para_LastRunTime' (Default value: 2017-06-08T13, Current value: 2022-06-24T08:43:05.261+00:00). On the right, the 'Activity Stream' panel shows a timeline of events for an integration instance, including scheduled runs, message processing, and data mapping.

Event	Timestamp	Description	Duration
Triggered a Scheduled Run	Set Jun 25 11:27:00 AM IST 2022		
Scheduled Run started executing	11:27:00:631 PM	118ms	
Message started processing	11:27:00:949 PM		
Message received by Assignment GlobalVariables	11:27:01:216 PM		
Message received by Data Mapper	11:27:01:217 PM	5ms	
Data Mapping completed with Message	11:27:01:222 PM	0ms	
Message received by Stage File Appendfile	11:27:01:222 PM	869ms	
Message received by Stage File Appendfile	11:27:02:091 PM	1ms	
While CheckResult	11:27:02:092 PM		1 Iteration

8. Click **View details**  to view specific details about the movement of the message payload in the activity stream. See [Track the Status of Integration Instances](#).
9. Click **Track instances** to access the Instances page filtered to show the completed instance.

The screenshot shows the 'Instances' page with a single completed instance listed. The instance details include the start time (2022-06-25T20:29:30.403+00:00), instance ID (g_T4vTFEE...), status (Succeeded), and duration (Received 5 minutes ago, Succeeded 5 minutes ago, Duration 12 seconds).

Primary Identifier	Instance Id	Status	Business Identifiers	Duration
start Time: 2022-06-25T20:29:30.403+00:00 Oracle Workday FTP ... 1.0.0	g_T4vTFEE...	Succeeded		Received 5 minutes ago Succeeded 5 minutes ago Duration 12 seconds

Based on the **Status** column value (for example, **Queued**, **Succeeded**, and others), the **Duration** column provides additional status details about the schedule instance. See [View the Status of Future Schedule Integrations](#).

Start and Pause an Integration Schedule

After you define a schedule, you must activate it. You can also pause (deactivate) a schedule, when needed. You can also submit schedules to run immediately.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the row of the integration on which the scheduled run is defined.
3. Click **Actions** , then select **Schedule**.

The Schedule and future runs page is displayed.

4. Click **Start** 
5. If you are a user with the ServiceAdministrator role, you can change the user with which to submit the schedule.
6. Click **Confirm** to start the schedule.

If you have defined scheduled parameters on this integration, the Schedule parameters page is displayed.

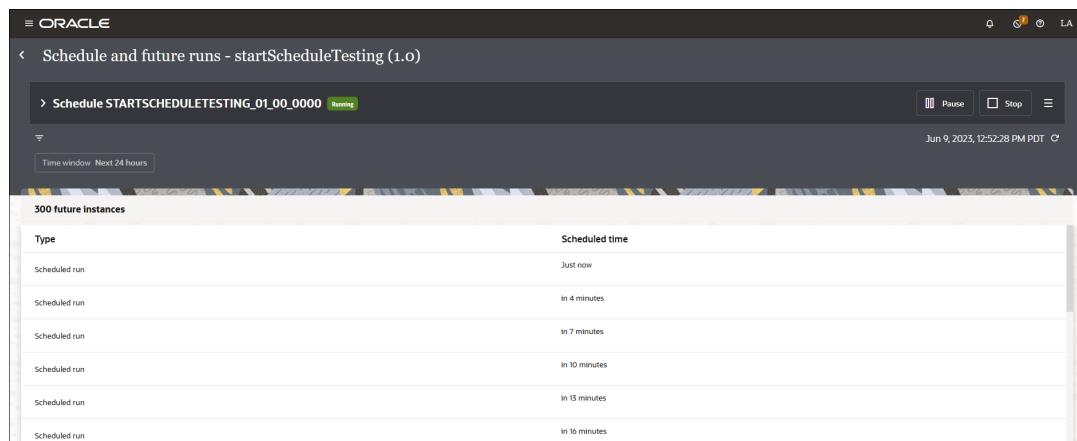
7. Make any necessary parameter value updates in the **New value** field, and click **Start Schedule**.

The **Start**  button is changed to **Stop** .

 **Note:**

If you want to pause the schedule run, click **Pause**  in the upper right corner, then click **Confirm** when prompted.

8. Click **Refresh**  to refresh the status of the schedule integration.



The **Pause** and **Stop** buttons are replaced with **Retry** and **Refresh** buttons if the schedule operation doesn't complete immediately after submission. In these cases, the status is displayed as **Stopping**, **Starting**, **Pausing**, or **Resuming**. The **Retry** option can be used to recover from these states. There must be a gap of 10 minutes between two retry attempts.

For example:



or



9. Click **Schedule menu** , then select **Track instances**.

The Instances page shows details about the run of the scheduled instance.

Instances																																		
Jun 9, 2023, 01:00:40 PM PDT C																																		
Time window Last 1 hour Integration STARTSCHEDULETESTING (STARTSCHEDULETESTING 01.00.0000) X Display instances Retained Sort by Last updated time Clear																																		
5 instances																																		
<table border="1"> <thead> <tr> <th>Primary identifier</th><th>Instance ID</th><th>Status</th><th>Business Identifiers</th><th>Duration</th></tr> </thead> <tbody> <tr> <td>startTime: 2023-06-09T19:59:52.613+00:00 startScheduleTesting 1.0.0</td><td>qIzOSQb_Ee...</td><td>Succeeded</td><td></td><td>Received Just now Succeeded Just now Duration 88 milliseconds</td></tr> <tr> <td>Primary: undefined startScheduleTesting 1.0.0</td><td>MqmweQcA...</td><td>Queued</td><td></td><td>Submitted Just now Scheduled in a minute</td></tr> <tr> <td>startTime: 2023-06-09T19:56:03.920+00:00 startScheduleTesting 1.0.0</td><td>PvU7Vgb_Ee...</td><td>Succeeded</td><td></td><td>Received 5 minutes ago Succeeded 5 minutes ago Duration 115 milliseconds</td></tr> <tr> <td>startTime: 2023-06-09T19:53:03.749+00:00 startScheduleTesting 1.0.0</td><td>1CYyFQb-Ee...</td><td>Succeeded</td><td></td><td>Received 8 minutes ago Succeeded 8 minutes ago Duration 104 milliseconds</td></tr> <tr> <td>startTime: 2023-06-09T19:50:04.551+00:00 startScheduleTesting 1.0.0</td><td>1STkBwb-Ee6...</td><td>Succeeded</td><td></td><td>Received 11 minutes ago Succeeded 11 minutes ago Duration 78 milliseconds</td></tr> </tbody> </table>					Primary identifier	Instance ID	Status	Business Identifiers	Duration	startTime: 2023-06-09T19:59:52.613+00:00 startScheduleTesting 1.0.0	qIzOSQb_Ee...	Succeeded		Received Just now Succeeded Just now Duration 88 milliseconds	Primary: undefined startScheduleTesting 1.0.0	MqmweQcA...	Queued		Submitted Just now Scheduled in a minute	startTime: 2023-06-09T19:56:03.920+00:00 startScheduleTesting 1.0.0	PvU7Vgb_Ee...	Succeeded		Received 5 minutes ago Succeeded 5 minutes ago Duration 115 milliseconds	startTime: 2023-06-09T19:53:03.749+00:00 startScheduleTesting 1.0.0	1CYyFQb-Ee...	Succeeded		Received 8 minutes ago Succeeded 8 minutes ago Duration 104 milliseconds	startTime: 2023-06-09T19:50:04.551+00:00 startScheduleTesting 1.0.0	1STkBwb-Ee6...	Succeeded		Received 11 minutes ago Succeeded 11 minutes ago Duration 78 milliseconds
Primary identifier	Instance ID	Status	Business Identifiers	Duration																														
startTime: 2023-06-09T19:59:52.613+00:00 startScheduleTesting 1.0.0	qIzOSQb_Ee...	Succeeded		Received Just now Succeeded Just now Duration 88 milliseconds																														
Primary: undefined startScheduleTesting 1.0.0	MqmweQcA...	Queued		Submitted Just now Scheduled in a minute																														
startTime: 2023-06-09T19:56:03.920+00:00 startScheduleTesting 1.0.0	PvU7Vgb_Ee...	Succeeded		Received 5 minutes ago Succeeded 5 minutes ago Duration 115 milliseconds																														
startTime: 2023-06-09T19:53:03.749+00:00 startScheduleTesting 1.0.0	1CYyFQb-Ee...	Succeeded		Received 8 minutes ago Succeeded 8 minutes ago Duration 104 milliseconds																														
startTime: 2023-06-09T19:50:04.551+00:00 startScheduleTesting 1.0.0	1STkBwb-Ee6...	Succeeded		Received 11 minutes ago Succeeded 11 minutes ago Duration 78 milliseconds																														

The following status values are displayed during the course of the schedule integration life cycle:

- Queued (Typically the state of an instance when it is scheduled to process sometime in the future.)
- Succeeded
- Errorred
- Aborted
- In progress
- Queued
- Paused (The schedule has been paused, meaning the instance is also paused. Unpausing the schedule moves the instance back to the waiting state.)
- Blocked
- Resubmitted

10. If necessary, schedule integration instances in the queued or paused state can be canceled by selecting **Abort**.

Instances														
Jun 9, 2023, 01:02:28 PM PDT C														
Time window Last 1 hour Integration STARTSCHEDULETESTING (STARTSCHEDULETESTING 01.00.0000) X Display instances Retained Sort by Last updated time Clear														
6 instances														
<table border="1"> <thead> <tr> <th>Primary identifier</th><th>Instance ID</th><th>Status</th><th>Business Identifiers</th><th>Duration</th></tr> </thead> <tbody> <tr> <td>Primary: undefined startScheduleTesting 1.0.0</td><td>gNxOYQcAE...</td><td>Paused</td><td></td><td>Submitted Just now Paused Just now</td></tr> </tbody> </table>					Primary identifier	Instance ID	Status	Business Identifiers	Duration	Primary: undefined startScheduleTesting 1.0.0	gNxOYQcAE...	Paused		Submitted Just now Paused Just now
Primary identifier	Instance ID	Status	Business Identifiers	Duration										
Primary: undefined startScheduleTesting 1.0.0	gNxOYQcAE...	Paused		Submitted Just now Paused Just now										
														

The activity stream provides details about the cancellation.

The screenshot shows the Instances page with two integration instances listed. The first instance, 'startScheduleTesting | 1.0', has a status of 'Aborted'. The second instance, 'startScheduleTesting | 1.0.0', has a status of 'Succeeded'. To the right, an activity stream window displays several milestones for the aborted instance, including 'Scheduled Run is queued' at 01:02:21.638 PM and 'Instance aborted by l2serviceadmin' at 01:03:38.896 PM.

This action moves the schedule integration instance to the aborted state. If this is a regularly scheduled integration (for example, it is scheduled daily at the same time), another instance gets created immediately for the next day's schedule and moved to the queued state.

If the schedule integration run is successful, the status value is updated to **Succeeded**.

11. Click **View**  to view schedule milestones such as the following in the activity stream.
 - Scheduled run is triggered: Indicates the time at which the request to trigger the schedule integration was submitted.
 - Scheduled run started processing: Indicates the time at which the submitted request started processing.
 - Schedule paused: Indicates the time at which a schedule was paused.
 - Schedule resumed: Indicates the time at which the schedule was resumed (from a paused state).
 - Processing completed successfully: Indicates the time at which schedule processing completed successfully.

Change the User Submitting an Integration Schedule

You can designate a specific user to control the schedule of an integration in the absence of the actual submitter. This enables integrations to be scheduled using a service account that does not go away if a user leaves the company or moves to another department. Only a user with the **ServiceAdministrator** role can change the submitter of a schedule.

1. Go to the Schedule and future runs page. See [Define the Integration Schedule](#).
2. Click **Actions** , then **Update submitter** to update the user that can submit the schedule. The Update submitter dialog is displayed.
3. Change the submitter of the schedule in the drop-down list.

This update is per a specific integration schedule, and not globally for all integration schedules that can be submitted by this user. You can globally change the submitter of integration schedules under **Settings > Schedule**. See [Globally Change the Submitter of Integration Schedules](#).

Details about the user who submitted the schedule for the integration are visible in the design-time audit logs on the Dashboard page. For example:

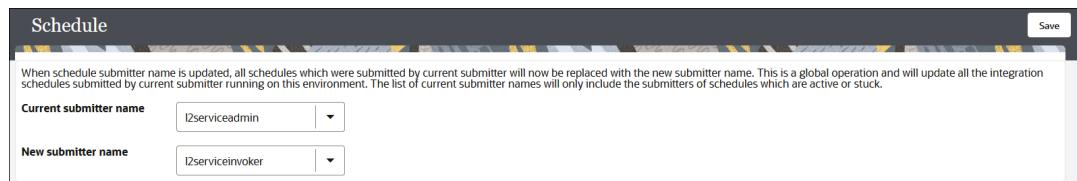
```
User icsadmin triggered integration_integration_name as user icsdeveloper
```

Child (co-located) integrations invoked from a parent schedule integration inherit the same user that submitted the schedule. See [Invoke a Child Integration from a Parent Integration](#).

Globally Change the Submitter of Integration Schedules

You can globally change all schedules submitted by the current user to another user. The list of current submitter names only include submitters of those schedules that are currently active or stuck. Only a user with the *ServiceAdministrator* role can perform this task.

1. In the navigation pane, click **Settings**, then **Schedule**.
2. From the **Current submitter name** list, select the submitter to replace.
3. From the **New submitter name** list, select the new submitter.



For this example, any schedules that the old **I2gen3serviceadmin** user submitted can now be submitted by the new **I2gen3serviceinvoker** user.

4. Click **Save**.

View Past and Present Schedule Integration Runs

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

1. In the navigation pane, click **Design**, then **Integrations**.
2. Go to the row of the integration on which the scheduled run is defined.
3. Click **Actions** , then select **Schedule**.

The Schedule and future runs page is displayed.

4. Click **Actions** , then select **Track instances**.
- The Instances page is displayed. By default, all completed integration runs are displayed.
5. Click **Filter**  to filter the display of integration runs.
 6. 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 in the activity stream.

View Future Schedule Integration Runs

You can view the status of future schedule integration runs.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the integration on which the scheduled run is defined.

- Click **Actions ...**, then select **Schedule**.

The Schedule and future runs page is displayed. Details about scheduled future runs are displayed. You can filter the display of integration runs. The page only displays manually submitted runs with a state of blocked, queued, ready, paused, or canceling. The time zone in which the schedule was created is also displayed.

Type	State	Scheduled Time
Scheduled Run	Yet To Start	in 4 minutes
Scheduled Run	Yet To Start	in 9 minutes
Scheduled Run	Yet To Start	in 14 minutes
Scheduled Run	Yet To Start	in 19 minutes
Scheduled Run	Yet To Start	in 24 minutes
Scheduled Run	Yet To Start	in 29 minutes

Note:

If you select a different time zone in another browser session, the **Scheduled time** field continues to show the time zone in which the schedule was created. All future runs process per this time zone, and not the time zone you specified in another browser session.

Edit an Integration Schedule

You can edit a schedule for an integration run.

- In the navigation pane, click **Design**, then **Integrations**.
- Hover over the row of the integration on which the scheduled run is defined.
- Click **Actions ...**, then select **Schedule**.

The Schedule and future runs page is displayed.

- Click **Edit**
- Edit the schedule. See [Define the Integration Schedule](#).
- Click **Save**.

Delete an Integration Schedule

You can delete a schedule for an integration. This action deletes all future runs of the schedule and stops any currently running schedule integrations.

- In the navigation pane, click **Design**, then **Integrations**.

2. Hover over the row of the integration on which the scheduler run is defined.
3. Click **Actions ...**, then select **Schedule**.
The Schedule and future runs page is displayed.
4. Click **Schedule menu** , then select **Delete schedule**.
5. Click **Confirm** when prompted. This deletes the schedule. Any scheduled jobs currently running are stopped.

Unlock an Integration Schedule

When a schedule is in edit mode and the browser fails, the schedule becomes locked, which prevents it from being edited. You can unlock an integration schedule to resume editing.

1. Log in as a user with the Administrator role.
2. In the navigation pane, click **Design**, then **Integrations**.
3. Click **Actions ...**, then select **Schedule**.
The Schedule and future runs page is displayed.
4. Click **Actions ...**, then select **Unlock schedule**.
5. Click **Unlock** when prompted. Note that unlocking an integration schedule may cause data loss.

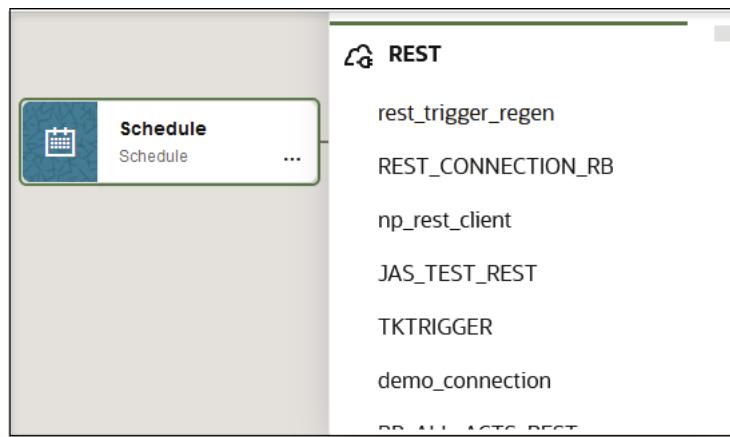
Convert a Schedule Integration to a REST Adapter-Triggered, Application Integration

You can convert a schedule integration to a REST Adapter-triggered, application integration. This eliminates the need to recreate the integration if you need to change the integration pattern from schedule to application.

Note the following restrictions:

- You cannot convert a schedule integration to a REST Adapter-triggered integration with multiple verbs.
- You cannot convert a schedule integration with assignments that have the fault data type.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Open the schedule integration you want to convert.
The integration canvas is displayed.
3. Click the Schedule icon, then click **Actions ...**, and then select **Replace**.
A list of available REST Adapter triggers is displayed.



4. Select the REST Adapter to use as the trigger connection in this integration.

This invokes the Adapter Endpoint Configuration Wizard.

5. Configure the REST Adapter in the wizard.

 **Note:**

Do not select the **Select to configure multiple resources or verbs** check box on the Basic Info page. This feature is not supported when converting schedule integrations to application integrations.

6. When wizard configuration is complete, click **Done**.

A message is displayed.

Clicking done will perform conversion to REST trigger and will clear tracking information, delete any assignment variables with fault data types and update any impacted actions. Errors and warnings may occur which will require manual correction. Are you sure you want to continue?

7. Review the message, then click **Convert**.

A conversion progress bar is displayed. When complete, the converted integration is displayed in the canvas with the REST Adapter trigger connection.

8. Investigate any errors and manually correct the impacted actions or invokes. For example, any scheduled parameters in the schedule integration are deleted during conversion to an application integration. This results in errors in any actions that reference those schedule parameters (for example, an assign or logger action).

Delete Scheduled or Ad-Hoc Runs

You can gracefully delete scheduled or ad-hoc runs in a number of states. Only a user with the administrator role can perform this activity. Any running instances triggered by these scheduled and ad-hoc runs are also terminated when you delete the run.

You can delete the following states for both scheduled and ad-hoc runs:

- Queued

- Ready
 - Running
 - Completed
 - Blocked
 - Hold
 - Paused
 - Schedule Ended
 - Error Auto Retry
 - Error Manual Recovery
1. In the navigation pane, click **Design**, then **Integrations**.
 2. Hover over the integration on which the scheduled run is defined.
 3. Click **Actions ...** and select **Schedule**.
The Schedule and future runs page is displayed. The state of each run is displayed.
 4. Click **Schedule menu** , then select **Delete schedule**.
 5. Click **Yes** when prompted to confirm your selection.
The scheduled run is removed from the Schedule and future runs page.

Monitor Schedule Integration Runs

You can pause and resume scheduled runs for an integration from the Monitor integrations page.

1. In the navigation pane, click **Observability**, then **Integrations**.

For schedule integrations, the  icon is displayed.
2. If you want to resubmit a run of a schedule integration, click **Actions ...**, then select **Run**.
3. If you want to pause scheduled runs, click **Actions ...**, then select **Pause schedule**.

Understand Schedule Integration Behavior When Using the Local Time Zone

Understand how schedule integrations are impacted by use of the local time zone. This section provides several use cases.

Daylight Savings Time Begins

Assume daylight savings begins in your environment on March 1 (time is moved forward from 1 AM to 2 AM) and your integration is scheduled to run at 1:30 AM. In this scenario, the time range between 1 AM and 1:59 AM does *not* exist and the schedule integration does *not* run on March 1 at 1:30 AM.

Daylight Savings Time Ends

Assume daylight savings time ends in your environment on November 1 (the time is rolled back from 2 AM to 1 AM) and your integration is scheduled to run at 1 AM. The schedule integration runs when expected at 1 AM. When 2 AM arrives and the time is rolled back to 1 AM, the schedule integration does *not* run again.

Create Event Integrations

You can create events that you publish and subscribe to in integrations in Oracle Integration.

Topics:

- [About Publishing and Subscribing to Events](#)
- [Create Events to Publish and Subscribe to in Integrations](#)
- [Publish Events in an Integration](#)
- [Subscribe to Events in an Integration](#)
- [Define Custom Header Filters to Trigger Integrations](#)

You can monitor integrations that have subscribed to events at runtime. See [Monitor Integrations That Subscribe to Events](#).

Note:

You can also create events to publish and subscribe to in integrations in a project. See [Create, Publish, and Subscribe to Events in a Project](#).

Want to see a video demo of publishing and subscribing to events in integrations?



About Publishing and Subscribing to Events

You can create and select events for publishing and subscribing to in integrations in Oracle Integration. You define the events in JSON-formatted or XML schema files. You can also define custom header-based filters in events. This section provides an overview of the event design life cycle and describes event restrictions.

- [Event Design Life Cycle](#)
- [Restrictions](#)

Event Design Life Cycle

The publish and subscribe feature enables you to decouple producers and subscribers. This decoupling enables you to define an event and start building your subscriber for the event before the event is published. You can create an event type (can be a producer, subscriber, or someone else). Creating an event type defines a contract, meaning that there's a contract to produce and subscribe.

The event design life cycle consists of three high-level steps. These steps are described using an event named Patient Moved as an example. References are provided to more specific documentation for performing these steps.

Step	Description	For Specific Details...
1 - Create an event	<p>Create a JSON-formatted or XML schema event on the Events page of Oracle Integration. For this example, the Patient Moved event contains the following JSON-formatted contents:</p> <pre>{ "patientId": "value1", "fromDept": "value1", "toDept": "value1" }</pre> <p>You can also define custom headers to which to add filters later when creating a subscribing integration. See Define Custom Header Filters to Trigger Integrations.</p>	See Create Events to Publish and Subscribe to in Integrations .
2 - Create an event publishing integration	<p>Create a publishing integration to publish the event:</p> <ol style="list-style-type: none"> 1. Create an integration. 2. Drag a Publish event action into the integration canvas. 3. Select Patient Moved as the event to publish in the Choose event panel. 	See Publish Events in an Integration .
3 - Create an event subscription integration	<p>Create a subscription integration to subscribe to the event:</p> <ol style="list-style-type: none"> 1. Select the Event integration pattern in the Create integration panel. This option creates an event-based integration pattern. 2. Select Patient Moved in the Choose event panel. 3. Run the publishing integration described in Step 2. This action triggers the event to be subscribed to by the subscription integration. 	See Subscribe to Events in an Integration .

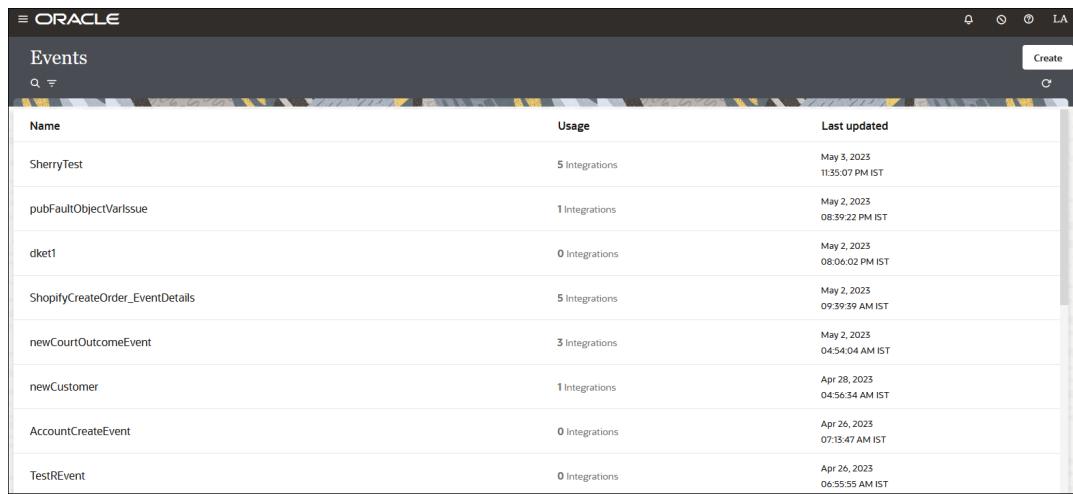
Restrictions

A maximum of 50 integrations can subscribe to events per service instance.

Create Events to Publish and Subscribe to in Integrations

You can create events that you publish in an integration and subscribe to in another integration. The contents of the event must be in JSON or XML schema format.

1. In the navigation pane, click **Design**, then **Events**.



The screenshot shows the Oracle Events interface. At the top, there's a search bar with placeholder text 'Events' and a 'Create' button. Below the header, a table lists eight events with columns for Name, Usage, and Last updated.

Name	Usage	Last updated
SherryTest	5 Integrations	May 3, 2023 11:35:07 PM IST
pubFaultObjectVarIssue	1 Integrations	May 2, 2023 08:59:22 PM IST
dket1	0 Integrations	May 2, 2023 08:06:02 PM IST
ShopifyCreateOrder_EventDetails	5 Integrations	May 2, 2023 09:39:39 AM IST
newCourtOutcomeEvent	3 Integrations	May 2, 2023 04:51:04 AM IST
newCustomer	1 Integrations	Apr 28, 2023 04:56:34 AM IST
AccountCreateEvent	0 Integrations	Apr 26, 2023 07:18:47 AM IST
TestREvent	0 Integrations	Apr 26, 2023 06:55:55 AM IST

2. Click **Create.**

The Define new event panel opens.

3. Enter a name (for this example, Patient Moved is entered) and optional description, then click **Continue.**

The Define event structure panel opens.

4. From the **Structure type list, select an option:**

- a. Select **JSON Sample** to upload a JSON file or drag and drop a JSON file into the box, then click **Create**. You can also manually edit the JSON sample provided to create your file contents.
- b. Select **XML Schema** to upload an XML schema file or drag and drop an XML schema file into the box, then click **Next**.

5. Click **Next.**

6. Click  to define a custom header name and description.

7. Click  to change the icon to a **Check Mark . This action enables the row to be edited.**

8. Enter a name and optional description, then click the **Check Mark  when complete.**

The event is displayed on the Events page.

 **Note:**

The **Search**  feature on the Events page only works on exact matches of names and is case sensitive.

9. Hover your cursor over the event, and click **View details  to view the file contents (for this example, a JSON-formatted file). The **Custom Headers** tab enables you to view any custom headers defined for this event. The **Subscribers** and **Publishers** tabs enable you to view the integrations that have subscribed to the event and published the event.**

The screenshot shows the Oracle Integration Cloud Service interface. On the left, there's a navigation pane with a search bar. The main area has a title 'Patient Moved' and a 'Details' section. Below that is a table with a single row labeled 'Patient Moved'. To the right of the table are tabs: 'Structure' (which is underlined in green), 'Custom Headers', 'Subscribers', and 'Publishers'. Under the 'Structure' tab, there's a 'Structure type:' dropdown set to 'JSON Sample'. Below it is a code editor window containing the following JSON:

```

1  {
2   "patientId": "value1",
3   "fromDept": "value1",
4   "toDept": "value1"
5 }
```

10. Perform other tasks, as necessary:

- Click **Edit** to edit the event. You can change the name or replace the contents of the file of the event.
- Click **Actions** to access options for cloning or deleting the event.

The event is available for publishing in an integration. See [Publish Events in an Integration](#).

Publish Events in an Integration

You can create an integration that publishes an event. Running the publishing integration enables subscribing integrations to subscribe to this event. This section provides an example of how to create a publishing integration.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Click **Create**, then select the integration pattern to use.

Note:

*Do not select **Event** as the integration pattern to publish the event. That pattern is for the integration that subscribes to the event. For this example, **Application** is selected to create an application integration.*

3. Add a trigger connection to the integration. For this example, a REST Adapter is selected and configured with the following details shown on the Summary page of the Adapter Endpoint Configuration Wizard:
 - Endpoints relative resource URI: **/patient**
 - Action to perform on the endpoint: **POST**
 - Query parameters specified: **patientId**, **fromDept**, and **toDept** (these are the parameters defined in the event file)

The screenshot shows a REST configuration interface with a navigation bar at the top. The navigation bar has 10 items: Basic Info, Response Configuration, Request Headers, Request Body, CORS Headers, Response Headers, Response Body, Operations, and Summary. Item 10, 'Summary', is highlighted. Below the navigation bar is a 'Summary' section containing a REST endpoint summary for a 'Patient' resource. At the bottom of the interface is a 'Endpoint Summary' section with details about a POST request to '/patient' with query parameters 'patientId', 'fromDept', and 'toDept'.

4. Drag a **Publish event** action into the integration canvas. See [Publish Events in an Integration with a Publish Event Action](#).
- The Choose event panel opens.
5. Select the event to publish, and click **Choose**. For this example, **Patient Moved** is selected. You can also create a new event from this panel by selecting **Define new event**.

The screenshot shows the 'Choose event' panel. The title is 'Choose event' and there is a 'Publish' button. A search bar contains the text 'Patient'. Below the search bar, three events are listed: 'patient-data' (PATIENT_DATA|1), 'newPatient' (NEWPATIENT|1), and 'Patient Moved' (PATIENT_MOVED). The 'Patient Moved' event is currently selected, as indicated by a blue selection bar underneath it.

The contents of the event are visible under the **View details** icon:

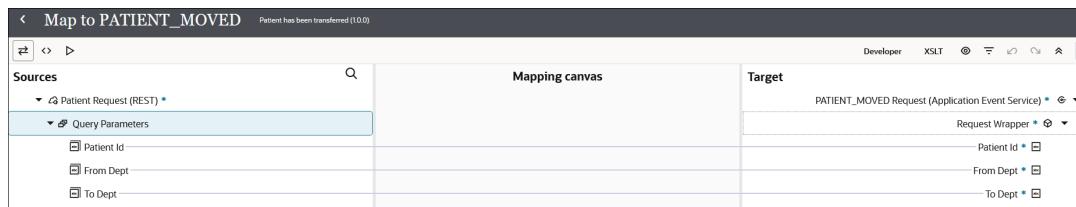
```
{
    "patientId": "value1",
```

```

    "fromDept": "value1",
    "toDept": "value1"
}

```

6. Click **Choose**, then click **Finish**.
7. Open the **Map** action that is automatically added above the **Publish event** action.
8. Map the source elements to the target elements. For this example, the three query parameters specified in the REST Adapter are mapped to the JSON event: **Patient Id**, **From Dept**, and **To Dept**.



9. Save and activate the integration.

You are now ready to create the integration that subscribes to this event.

Subscribe to Events in an Integration

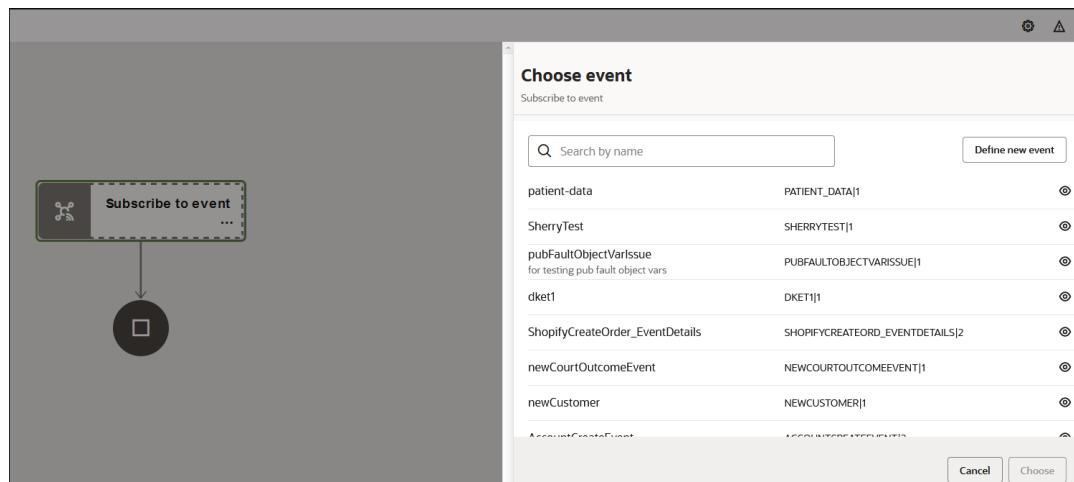
You can create an integration that subscribes to published events. The subscribing integration event is triggered when the integration publishing the event is run. Multiple integrations can subscribe to the same published event. This section provides an example of how to create a subscription integration.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Click **Create**, then select **Event**.
3. Complete the fields in the Create integration panel to create the event-based integration. See [Create an Integration](#).

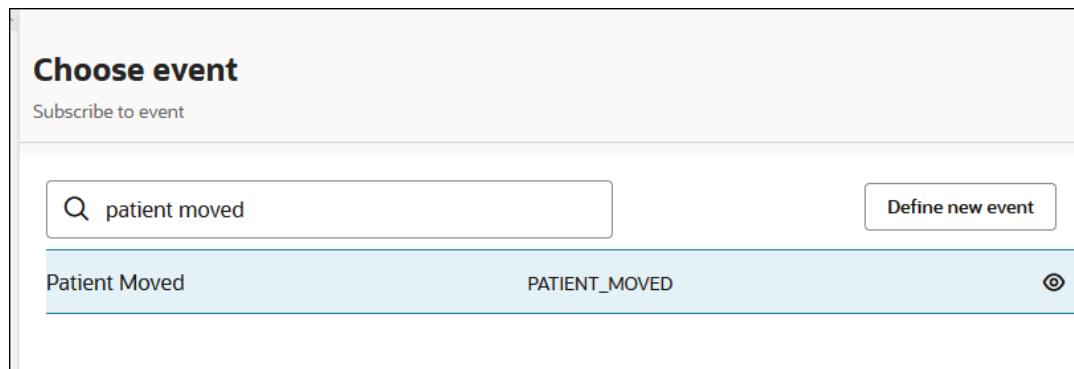
A **Subscribe to event** trigger is created and the Choose event panel opens for you to select the event to which to subscribe.

Note:

The selected event becomes the trigger for this integration. You do *not* drag an adapter connection into the integration canvas in event-based integrations to use as a trigger.



4. Select the event. For this example, **Patient Moved** is selected. If necessary, you can also define a new event in this panel by selecting **Define new event**.



5. Click **View details** to view the JSON contents of the event. For this example:

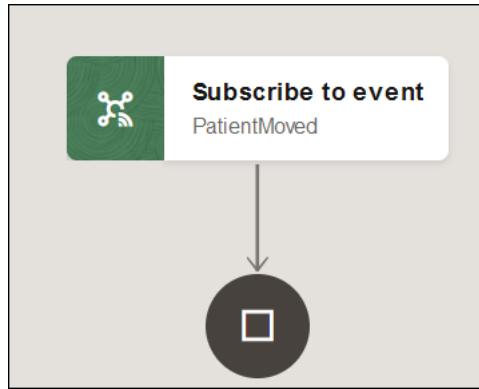
```
{
    "patientId": "value1",
    "fromDept": "value1",
    "toDept": "value1"
}
```

6. Click **Choose this event** or click <, then click **Choose**.

You can also define filters on any custom headers in the event. See [Define Custom Header Filters to Trigger Integrations](#).

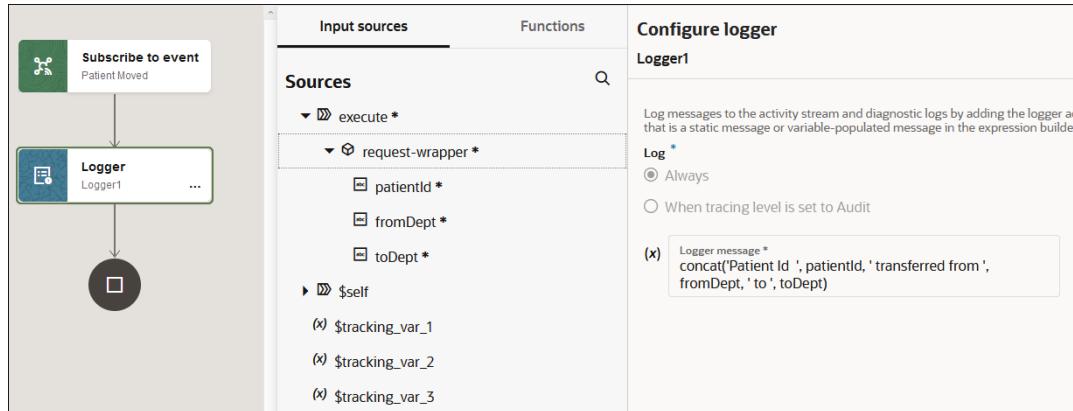
7. Click **Finish**.

The **Subscribe to event** trigger is updated to include the event that you selected.



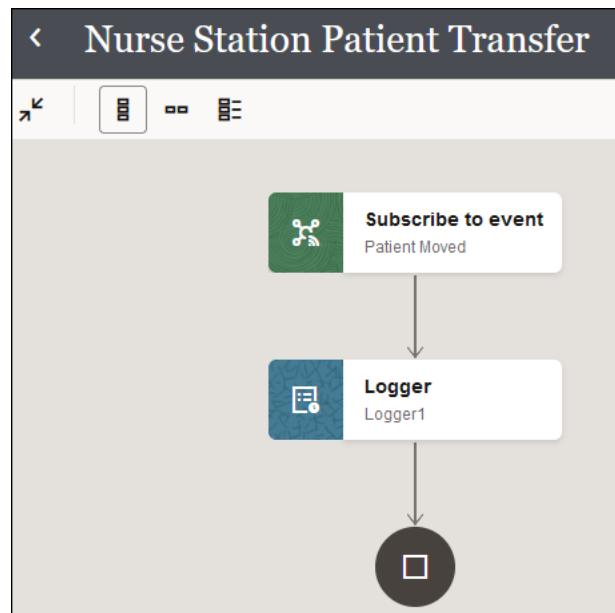
- Design the remaining parts of the integration. For this example, a logger action is added to the integration canvas and the message is configured to include the parameters defined in the event.

```
concat('Patient Id ', patientId, ' transferred from ', fromDept, ' to ',
toDept)
```



- Define a business identifier for the integration.

The integration looks as follows:



10. Save and activate the integration. This integration is now ready to subscribe to the **Patient Moved** event.
11. Run the integration that publishes the **Patient Moved** event. Once this integration is run, the event in the subscribing integration is triggered.
 - a. In the navigation pane, click **Design**, then **Integrations**.
 - b. Hover over the publishing integration, and select **Actions ...**, and then **Run**.
 - c. Specify values for the three parameters in the event file, and click **Run**.

Configure and run
Configure request properties, then run the integration and verify the response. [Click here to read more.](#)

Request

POST /ic/api/integration/v1/flows/rest/PATIE_TRANS_FROM_SURGE_TO_ICU/1.0/patient?patientId=0006&fromDept=Surgery&toDept=ICU

URI parameters	Headers	Body	cURL	Integration properties
<input checked="" type="checkbox"/> patientId	0006			
<input checked="" type="checkbox"/> fromDept	Surgery			
<input checked="" type="checkbox"/> toDept	ICU			

The activity stream opens and indicates that the **Patient Moved** event was published.

The screenshot shows the 'Activity stream' panel with the following details:

- Instance Id:** XB08Be0REe2nlav5R8neMA
- Tracing level:** Production
- Log Entries:**
 - Sun May 07 12:57:13.400 PM PDT 2023, 110ms: Wire Message dequeued and processed by Trigger Patient
 - 12:57:13.510 PM, 39ms: Message started processing
 - 12:57:13.549 PM, 1s 900ms: Data Mapping completed
 - 12:57:15.449 PM: Publish PATIENT_MOVED
 - 12:57:15.449 PM, 1s 990ms: Message publishing started by Publish PATIENT_MOVED
 - 12:57:17.439 PM: Message published by Publish PATIENT_MOVED

- In the navigation pane, click **Observability**, then **Integrations**.

The integration that subscribed to the event was triggered and an instance was created and processed successfully.

The screenshot shows the 'Monitor integrations' page with the following details:

- Total instance count:** 695
- Filter Options:** Time window (Last 1 hour), Status (Active), Display integrations (All), Sort by (Last update)
- Last updated:** May 9, 2023, 01:52:37 AM IST
- Table Headers:** Name, Received, Processed, Succeeded, Errored, Aborted, Last scheduled by
- Table Data:**

Name	Received	Processed	Succeeded	Errored	Aborted	Last scheduled by
Nurse Station Patient Transfer (1.0) Last message: Just now	1	1	1	0	0	

- In the navigation pane, click **Observability**, then **Instances**.

- Hover over the business identifier for the integration instance and click **View details** to open the Activity stream panel.

The subscribing integration successfully subscribed to the event. The logger action message in the activity stream shows that the patient transfer occurred.

Patient ID 0006 was transferred from surgery to ICU.

The screenshot shows two main panels. On the left, the 'Instances' panel displays a table of 14 instances. The columns include Primary Identifier, Instance Id, Status, Business Identifiers, and Duration. One instance is highlighted: 'patientId: 0006' with status 'Succeeded'. On the right, the 'Activity stream' panel shows a timeline of events for instance ID 0QKMjO0Tee2Xm-OHmSc1Zw. The events are:

- Sun May 07 01:48:512 PM PDT 2023, 35ms: Message received by Subscribe
- 01:48:547 PM, 1ms: Message started processing
- 01:48:548 PM: Logger Logger1
- Patient ID 0006 transferred from Surgery to ICU
- 01:48:548 PM: Processing completed successfully

Retain an Event Subscription During Integration Deactivation

You can retain an event subscription when deactivating an integration that subscribes to published events. This action enables you to begin subscribing to published events again when you reactivate the integration.

Note:

For subscribing integrations created in release 23.10 and later, the **Delete event subscription** check box is visible. For subscribing integrations created prior to release 23.10, there is no check box and you do not have the ability to retain events when deactivating an integration.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Go to the row of the subscribing integration you want to deactivate.
3. Click **Deactivate** to deactivate the integration.

You can select whether or not to delete the event subscription.



Event subscription behavior is based on what you do with the **Delete event subscription** check box.

4. If you do not select **Delete event subscription**, and click **Deactivate**.

Note:

Events expire after 24 hours. Therefore, only events published within one day of the next activation are delivered.

- a. In the navigation pane, click **Observability**, then **Subscriptions**.
- b. Note that the deactivated integration appears with the words **Configured** in the **Subscriber status** column. If you reactivate the integration, it begins subscribing to events again.
- c. If you want to unsubscribe, hover over the row of the integration, and click **X**.



5. If you select **Delete event subscription**, and click **Deactivate**.
 - a. In the navigation pane, click **Observability**, then **Subscriptions**.
 - b. Note that the deactivated integration does not appear. If you reactivate the integration, it does *not* subscribe to events again.

Define Custom Header Filters to Trigger Integrations

You can define custom headers in events, select those events and map request payload elements to the custom headers when creating a publishing integration, and then define filters on those headers when creating a subscribing integration. If the filter condition defined is satisfied at runtime, the subscribing integration is triggered. If the filter condition is not satisfied, the integration is not triggered.

- [Capabilities](#)
- [Restrictions](#)
- [Use Case Overview](#)
- [Create an Event with a Custom Header in a Project](#)
- [Create an Integration to Publish the Event in a Project](#)
- [Create an Integration with No Filter to Subscribe to the Event in a Project](#)
- [Create an Integration with a Name Filter to Subscribe to the Event](#)
- [Create an Integration with an Age Filter to Subscribe to the Event](#)
- [Run the Publishing Integration and Track the Created Instances](#)
- [Run the Publishing Integration Again and Track the Created Instances](#)

Capabilities

- You define custom headers when creating an event.

- You define custom header values in the mapping of a publish event action. You can define the mapping from payload data to a custom header to enable payload data to participate in filtering.
- You define filters on the custom headers when creating a subscribing integration. Subscribers can filter events based on fields within the payload if those fields are mapped to the custom headers. If the filter condition you define is satisfied during runtime, the subscribing integration is triggered.
- You define filters using JQ expressions.

```
"filter-def": ".customheader1==\"someValue\""
```

JQ is a lightweight and flexible command line JSON processor primarily used for extracting and manipulating JSON data. JQ takes an input and produces an output. See [jq 1.7 Manual](#) and [JQ Expression Syntax for Creating Filters](#).

Restrictions

- Only JSON-formatted filters are supported.
- When defining a filter condition, only the code editor is supported. You cannot drag a file into the page.
- You can define 10 custom headers per event.
- Enter all functions in lower case.
- When defining a filter condition, only JQ expression filters are supported.
- The following JQ function calls are supported:
 - any
 - all
 - ascii_downcase
 - ascii_upcase
 - tonumber
 - tostring
 - min
 - max
 - contains
 - startswith
 - endswith
 - ltrimstr
 - rtrimstr
 - length
- The following JQ keywords/filter expressions are not supported:
 - '.' (A single dot alone is not supported for security reasons.)
 - while
 - for

- repeat
- input/inputs/stderr
- debug
- input_filename
- input_line_number
- recurse
- foreach
- until
- JQ-based Lambda function definitions and their executions as part of a filter definition are not supported.

Use Case Overview

The use case in this section walks you through creating an event with custom headers, selecting the event and mapping the custom headers in a publishing integration, and adding different filter conditions to the headers in several subscribing integrations. In this use case:

1. Event **MLFilterEvent** is created with the following event and custom header content:

Event Structure	Custom Headers
<pre>{ "patientIdentifier" : "12345679", "illness" : "1", "name": "min", "age":30, "weight":100 }</pre>	<pre>ptname ptage ptweight</pre>

2. Publishing integration **MLFilterPub** is designed to publish the event **MLFilterEvent**. This integration consists of the following connection and actions:

- A REST Adapter trigger is configured with the following JSON-formatted request payload:

```
{
    "patientIdentifier" : "12345679",
    "illness" : "1",
    "name" : "min",
    "age" : 30,
    "weight" : 100
}
```

- A mapper is configured to map patient details elements in the source request payload to target patient details elements, target standard header elements, and target custom header elements.
- A log action is configured to log standard and custom header details in the activity stream.

3. The following subscribing integrations are designed to subscribe to the **MLFilterEvent** event:
- Subscribing integration **MLFilterSubNoFilter** is designed *without* adding any filters.
 - Subscribing integration **MLFilterNameSub** is designed to be triggered only if the following custom header filter condition is satisfied.

```
{
  "type": "jq_filter",
  "filter-def": ".ptname==\"John\""
}
```

- c. Subscribing integration **MLFilterAgeSub** is designed to be triggered only if the following custom header filter condition is satisfied.

```
{
  "type": "jq_filter",
  "filter-def": ".ptage|length!=0 and tonumber >= 18"
}
```

4. The three subscription integrations are activated: **MLFilterSubNoFilter**, **MLFilterNameSub**, and **MLFilterAgeSub**.
5. When publishing integration **MLFilterPub** is activated, the request payload body values are specified, and the integration is run, the following occurs:
- Subscribing integration **MLFilterSubNoFilter** is always triggered and consumes the event. Because no custom header filters are defined on this subscribing integration, there are no restrictions on it being triggered.
 - Subscribing integration **MLFilterNameSub** is only triggered if "name": "John" is set in the request payload body.

```
{
  "patientIdentifier" : "123456790",
  "illness" : "1",
  "name": "John",
  "age":30,
  "weight":100
}
```

- Subscribing integration **MLFilterAgeSub** is only triggered if age is greater than or equal to 18. For this example, "age": "30" is set in the request payload body.

```
{
  "patientIdentifier" : "123456790",
  "illness" : "1",
  "name": "John",
  "age":30,
  "weight":100
}
```

Let's now design and run the use case.

Create an Event with a Custom Header in a Project

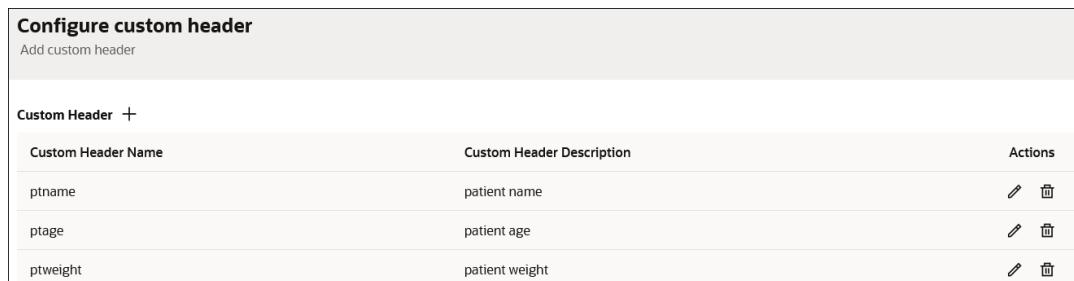
- In the navigation pane, click **Projects**.

2. Create a project.
3. Scroll down to the **Events** section.
4. Click **Add** if no events currently exist or **+** if events already exist.
5. Enter a name (for this example, named **MLFilterEvent**) and description, and click **Continue**.
6. Define the event structure, and click **Continue**. For this event, the following JSON-formatted event is defined:

```
{
    "patientIdentifier" : "12345679",
    "illness" : "1",
    "name": "min",
    "age":10,
    "weight":100
}
```

7. Click **+** to add a custom header to the event.
8. Click **Edit**  to change the icon to a **Check Mark ✓**. This action enables the row to be edited.
9. Enter a name and optional description, then click the **Check Mark ✓** when complete. Note the following custom header naming conventions:
 - Can be lower-case letters (a-z) and numbers (0-9).
 - Cannot be longer than 20 characters.
 - Cannot be a duplicate of another custom header.
 - Must not be a reserved cloud events attribute (for example, data). If you use a reserved name, it does not pass validation.

For this example, the following custom headers are added.



The screenshot shows a table titled 'Custom Header +' with three rows. The columns are 'Custom Header Name', 'Custom Header Description', and 'Actions'. The rows contain the following data:

Custom Header Name	Custom Header Description	Actions
ptname	patient name	 
ptage	patient age	 
ptweight	patient weight	 

Create an Integration to Publish the Event in a Project

1. Return to the project details page.
2. Go to the **Integrations** section and add an integration.
3. Select an application integration.

4. Add a trigger connection. For this example, a REST Adapter connection is defined to include the following JSON-formatted request payload:

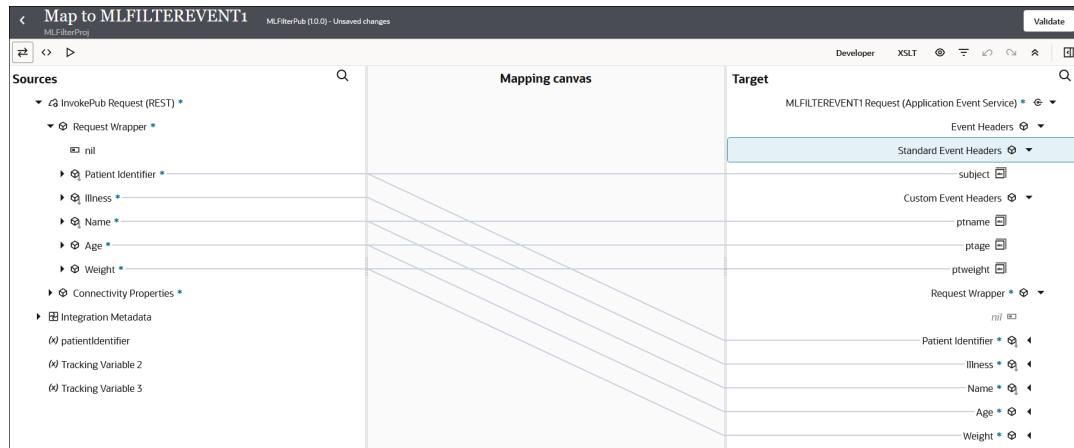
```
{
  "patientIdentifier" : "12345679",
  "illness" : "1",
  "name" : "min",
  "age" : 30,
  "weight" : 100
}
```

5. Click **+** below the REST Adapter and drag a **Publish event** action into the integration.
 6. Select the event in which you defined the custom headers (**MLFilterEvent**), and click **Finish**.

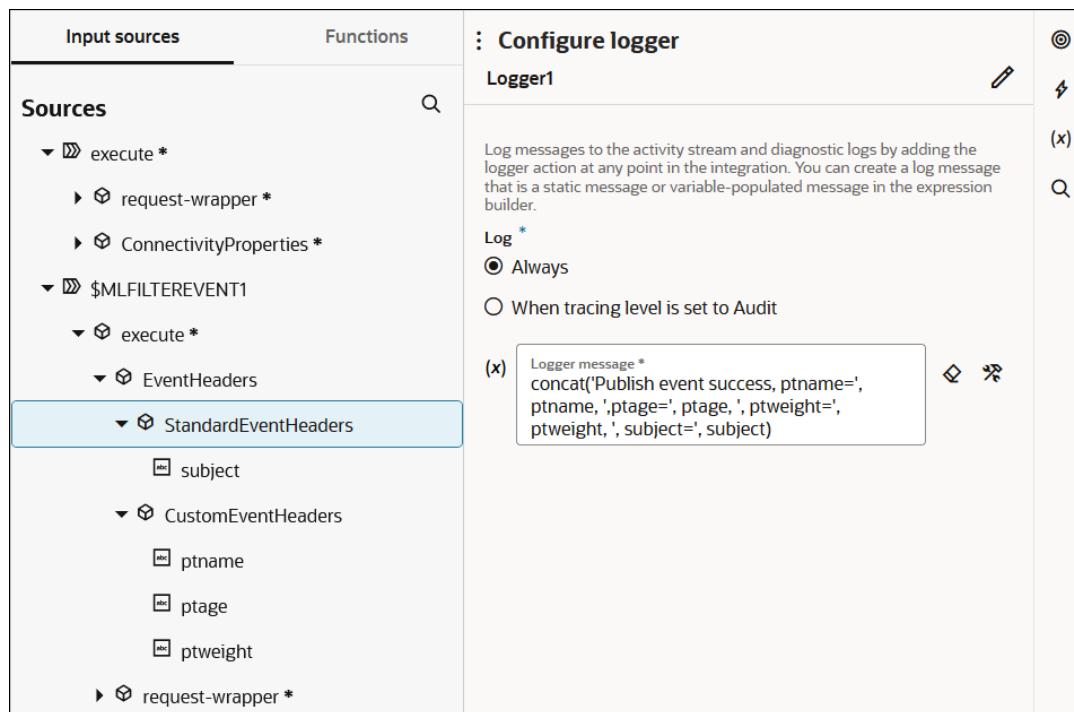


7. Open the mapper for the **Publish event** action and define the mappings:
- Map the **Request Wrapper** source elements to the **Request Wrapper** target elements. For this example:
 - Patient Identifier** to **Patient Identifier**
 - Illness** to **Illness**
 - Name** to **Name**
 - Age** to **Age**
 - Weight** to **Weight**
 - Expand the target **Event Headers** section to show **Standard Event Headers** and **Custom Event Headers**.
 - Expand the target **Standard Event Headers** and map the source **Patient Identifier** to the target **Subject**.
 - Expand the target **Custom Event Headers** and map the following request payload elements to the target custom headers you created:
 - Name** to **ptname**
 - Age** to **ptage**
 - Weight** to **ptweight**

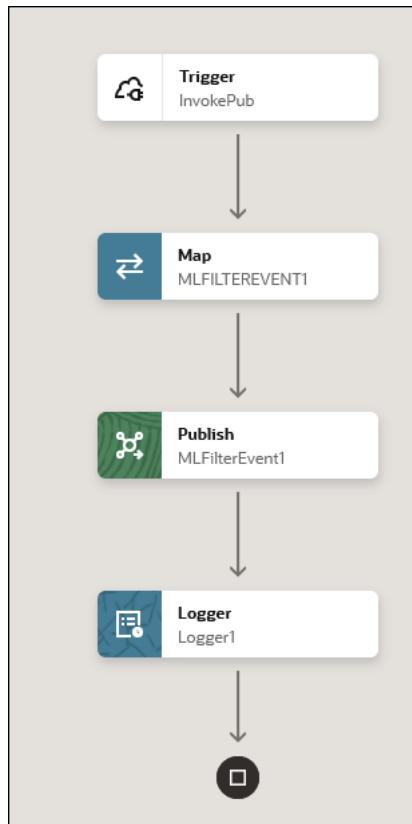
When complete, the mappings look as follows:



8. Validate the mapper and return to the integration canvas.
9. Add a log action to log the standard and custom headers.



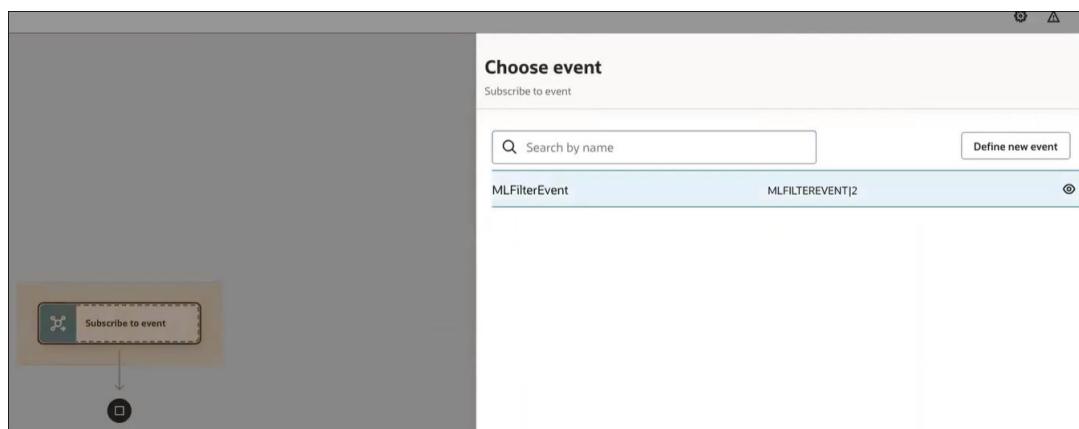
10. Save the log action.
- The completed integration looks as follows:



Let's now create integrations to subscribe to the event and demonstrate how filters work.

Create an Integration with No Filter to Subscribe to the Event in a Project

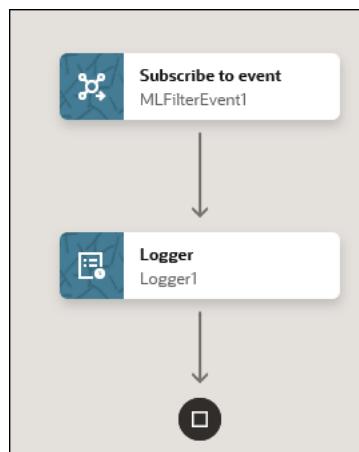
1. Create an event integration (for this example, named **MLFilterSubNoFilter**).
2. Select the **MLFilterEvent** event, and click **Choose**.



3. Click **Finish**.
4. Add a log action to log the standard and custom header details.

The screenshot shows the Oracle Integration Designer interface. On the left, there's a tree view under 'Input sources' labeled 'Sources' containing various event headers like 'execute*', 'EventHeaders' (with 'StandardEventHeaders' and 'subject'), 'CustomEventHeaders' (with 'ptweight', 'ptage', and 'ptname'), and 'request-wrapper*'. On the right, a detailed configuration panel for a 'Logger1' function is shown. It has sections for 'Logger message*' containing the expression `concat('Received event', ', ptname=', ptname, ', ptage=', ptage, ', ptweight=', ptweight)`, 'Log *' (radio button set to 'Always'), and 'When tracing level is set to Audit'. A note at the bottom says 'Typing is only available for literals and numbers in designer view. You can also toggle to developer view for full editing capabilities.'

The completed integration looks as follows:



5. Activate the integration.

Create an Integration with a Name Filter to Subscribe to the Event

1. Create an event integration (for this example, named **MLFilterNameSub**).
2. Select the **MLFilterEvent** event, and click **Choose**.

The screenshot shows the Oracle Integration Designer interface. On the left, a 'Subscribe to event' step is visible. On the right, a 'Choose event' dialog box is open, titled 'Choose event' with the subtitle 'Subscribe to event'. It contains a search bar 'Search by name' and a 'Define new event' button. Below the search bar, two event names are listed: 'MLFilterEvent' and 'MLFILTEREVENT|2'. The 'MLFilterEvent' entry is highlighted.

3. Click **Add Filter**.



4. Manually enter the filter condition in the code editor. The only supported filter type is `jq_filter`. For this example, `ptname` (patient name) is set to `John`. If this filter condition is satisfied at runtime, this integration is triggered.

```
{
  "type": "jq_filter",
  "filter-def": ".ptname==\"John\""
}
```

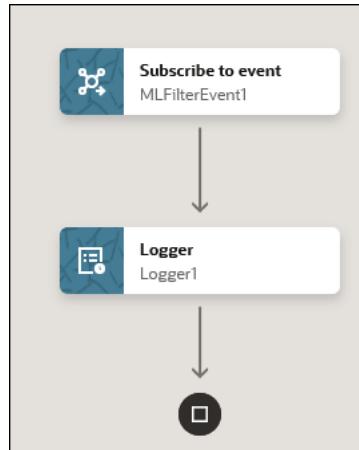
The screenshot shows the 'Configure Event Filter' screen. It has a header 'Configure Event Filter' and a sub-header 'Event Filter'. Below that is a note: 'To add event filtering rule Drag & Drop a JSON file or use code editor below.' and an example: 'Event filter sample: {"type": "jq_filter", "filter-def": ".customheader!=\"Asia\""}'. There is a 'Upload JSON file' section with a placeholder 'Select a file or drop one here'. Below that is a code editor containing the following JSON:

```

1 {
2   "type": "jq_filter",
3   "filter-def": ".ptname==\"John\""
4 }
```

5. Click **Validate**, then click **Finish**.
6. Add a log action with a message.

The completed integration looks as follows:



7. Activate the integration.

Create an Integration with an Age Filter to Subscribe to the Event

1. Create an event integration (for this example, named **MLFilterAgeSub**).
2. Select the **MLFilterEvent** event, and click **Choose**.

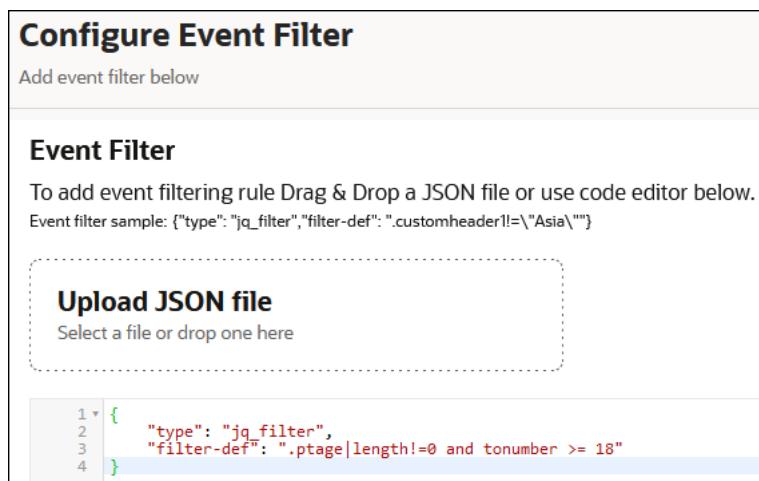
The completed integration looks as follows:

- Click **Add Filter**.



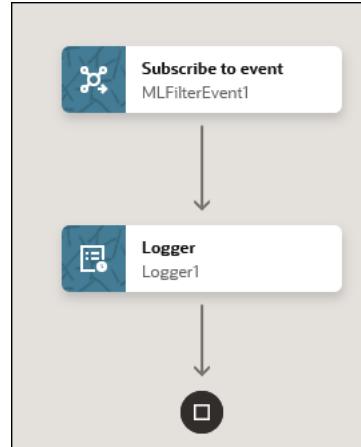
- Manually enter the filter in the code editor. For this example, `ptage` (patient age) is set to 18 or older. If this filter condition is satisfied at runtime, this integration is triggered.

```
{
    "type": "jq_filter",
    "filter-def": ".ptage|length!=0 and tonumber >= 18"
}
```



- Click **Validate**, then click **Finish**.
- Add a log action to log age details.

The completed integration looks as follows:



7. Activate the integration.

Run the Publishing Integration and Track the Created Instances

1. Activate the publishing integration.
2. Hover over the publishing integration, then click **...** and select **Run**. Under **Body**, the request payload is displayed.
3. Change **name** to **John** and **age** to **10**, then click **Run**.

```
{
  "patientIdentifier": "12345679",
  "name": "John",
  "weight": 100,
  "age": 10,
  "illness": "1"
}
```

4. Expand key milestones in the activity stream for the publishing integration instance to see details.

```

02:48:19.468 PM
Publish MLFILTEREVENT1_REQUEST

02:48:19.468 PM, 69ms
Message publishing started by Publish MLFILTEREVENT1_REQUEST

Cloud event id: xyh8x0bhEe-fKM-xBa6VlQ
Cloud event source:
com.oracle.oic.integrationInstance.integration:ocid1.integrationinstance.oc1.phx.ama
aaaaagokyqtya5bweujjcgsuhw4buioopfmwkfaiuev3ohl5spijnfgq:MLFILTERPUB:01.00.0000:MLF
ILTERPROJ
Cloud event type: MLFILTEREVENT
Cloud event version: 2

02:48:19.545 PM
Logger Logger1

Publish event success, ptname=John,ptage=10, ptweight=100, subject=12345679

```

5. Return to the project details page.

6. Click **Observe**, then **Instances**.

The following integration instances are displayed.

- **MLFilterSubNoFilter**: The subscribing integration with no filter. This integration is triggered because no filter is set.
- **MLFilterNameSub**: The subscribing integration with the filter of `name` set to `John`. This integration was triggered because the filtering condition was satisfied.
- **MLFilterPub**: The publishing integration for the event.

Jul 20, 2024, 02:55:26 PM PDT				
Primary identifier	Instance ID	Business Identifiers	Status	Duration
patientIdentifier: 12345679 MLFilterSubNoFilter 1.0.0	yErcvUbhEe-fKM-x...		Succeeded Succeeded Duration	7 minutes ago 7 minutes ago 90 milliseconds
patientIdentifier: 12345679 MLFilterNameSub 1.0.0	yEmkOObhEe-fKM....		Succeeded Succeeded Duration	7 minutes ago 7 minutes ago 81 milliseconds
patientIdentifier: 12345679 MLFilterPub 1.0.0	xxATTObhEe-TZzdK...		Succeeded Succeeded Duration	7 minutes ago 7 minutes ago 238 milliseconds

The subscribing integration **MLFilterAgeSub** was not triggered because the filtering condition was not satisfied.

7. Open the activity stream and expand key milestones for **MLFilterSubNoFilter**.

Activity stream

Instance ID: yErcvUhbEe-fKM-xBa6VIQ

Tracing level: **Production**

Sat Jul 20 02:48:21.371 PM PDT 2024
Message received by Subscribe

Cloud event id: xyh8x0bhEe-fKM-xBa6V1Q
Cloud event source:
com.oracle.oic.integrationInstance.integration:ocid1.integrationinstance.oc1.phx.amaaaaaagok
yqtya5bweujcgsuhw4buioopfmwkfaieuve3oh15spijnfgq:MLFILTERPUB:01.00.0000:MLFILTERPROJ
Cloud event type:
application.oc1.phx.amaaaaaasmfv4eaanxxzx6cqjufxijl76lkmevzzcv3nilw7oqhwbmdq6wzq.MLFILTERPROJ.MLFILTEREVENT
Cloud event version: 2

02:48:21.406 PM
Message started processing

02:48:21.460 PM
Logger Logger1

Received event, ptname=, ptag=, ptweight=

02:48:21.461 PM
Processing completed successfully

8. Open the activity stream and expand key milestones for **MLFilterNameSub**.

The screenshot shows the Activity stream interface for an integration instance. The instance ID is yEmkOObhEe-fKM-xBa6VIQ. The tracing level is set to Production. The timeline of events is as follows:

- Message received by Subscribe**: Occurred at Sat Jul 20 02:48:21.363 PM PDT 2024. This event includes a detailed log message with a long Cloud event ID and source information.
- Message started processing**: Occurred at 02:48:21.404 PM.
- Logger Logger1**: Occurred at 02:48:21.444 PM. This event is associated with the message starting processing.
- Received event**: A placeholder message indicating an event was received.
- Processing completed successfully**: Occurred at 02:48:21.444 PM.

Run the Publishing Integration Again and Track the Created Instances

1. Return to the project details page.
2. Hover over the publishing integration, then click **...** and select **Run**. Under **Body**, the request payload is displayed:
3. Leave name set to John, change age to 30, optionally change the patientIdentifier to another value, and then click **Run**.

```
{
  "patientIdentifier": "123456890",
  "name": "John",
  "weight": 100,
  "age": 30,
  "illness": "1"
}
```

4. Expand key milestones in the activity stream for the publishing integration instance to see details.

03:23:57.641 PM
Publish MLFILTEREVENT1_REQUEST

03:23:57.641 PM, 138ms
Message publishing started by Publish MLFILTEREVENT1_REQUEST

Cloud event id: wZvLqkbmEe-wV7PhAaURtg
Cloud event source: com.oracle.oic.integrationInstance.integration:ocid1.integrationinstance.oc1.phx.amaaaaaagokyqtya5bweujjcgsuhw4buioopfmwkfaiuev3ohl5spijngqq:MLFILTERPUB:01.00.0000:MLFILTERPROJ
Cloud event type: MLFILTEREVENT
Cloud event version: 2

03:23:57.787 PM
Logger Logger1

Publish event success, ptname=John,ptage=30, ptweight=100, subject=123456890

5. Return to the project details page.
6. Click **Observe**, then **Instances**.
The following integration instances are displayed.
 - **MLFilterSubNoFilter**: The subscribing integration with no filter. This integration is triggered because no filter is set.
 - **MLFilterAgeSub**: The subscribing integration with the filter of `age` set to 30. This integration was triggered this time because the filtering condition was satisfied.
 - **MLFilterNameSub**: The subscribing integration with the filter of `name` set to `John`. This integration was triggered again because the filtering condition was satisfied.
 - **MLFilterPub**: The publishing integration for the event.

Jul 20, 2024, 03:30:16 PM PDT				
Time window Last 1 hour	Display instances Retained	Sort by Last updated time		
Primary identifier	Instance ID	Business Identifiers	Status	Duration
patientIdentifier: 123456890 MLFilterSubNoFilter 1.0.0	won8M0bmEe-fKM...		Succeeded	Received 7 minutes ago Succeeded 7 minutes ago Duration 122 milliseconds
patientIdentifier: 123456890 MLFilterAgeSub 1.0.0	wonVIEbmEe-fKM....		Succeeded	Received 7 minutes ago Succeeded 7 minutes ago Duration 120 milliseconds
patientIdentifier: 123456890 MLFilterNameSub 1.0.0	wonVH0bmEe-fKM...		Succeeded	Received 7 minutes ago Succeeded 7 minutes ago Duration 77 milliseconds
patientIdentifier: 123456890 MLFilterPub 1.0.0	wW9IGkbmEe-6GW...		Succeeded	Received 7 minutes ago Succeeded 7 minutes ago Duration 438 milliseconds

7. Open the activity stream and expand key milestones for each instance. For this example, **MLFilterAgeSub** is expanded.

The screenshot shows the Oracle Cloud Activity Stream interface. At the top, it displays "Activity stream" and "Instance ID: wonVIEbmEe-fKM-xBa6VIQ". The tracing level is set to "Production". The timeline shows the following events:

- Message received by Subscribe:** Occurred at 03:23:59.201 PM PDT 2024. The event details include a long Cloud event ID and source information.
- Message started processing:** Occurred at 03:23:59.247 PM. The log message is "Received event, age= [redacted]".
- Logger Logger1:** Occurred at 03:23:59.321 PM. The log message is "Processing completed successfully".

JQ Expression Syntax for Creating Filters

This section describes common JQ syntax elements you can use to create filter conditions.

JQ Filter	Description	JSON Sample	Result
[.[]]	Selects all elements of an array.	[1, 2, 3, 4, 5]	[1, 2, 3, 4, 5]
[.[] .field]	Selects a specific field from each object in an array.	[{"field": "value1"}, {"field": "value2"}]	["value1", "value2"]
[.[] select(.field == "value")]	Filters elements where a specific field equals a value.	[{"field": "value1"}, {"field": "value2"}]	[{"field": "value1"}]

JQ Filter	Description	JSON Sample	Result
<code>map(has("field"))</code>	Extracts the results of a specific function applied.	<pre>[{"field": "value1"}, {"field": "value2"}]</pre>	[true, true]
<code>[.[] .field select(. == "value")]</code>	Filters elements of an array where a specific field equals a value.	<pre>[{"field": "value"}, {"field": "value1"}]</pre>	["value"]
<code>[.[] select(length > 5)]</code>	Filters elements with a length greater than 5.	<pre>["apple", "banana", "orange", "grape"]</pre>	["banana", "orange"]
<code>[.[] select(.field1 == "value" and .field2 > 10)]</code>	Selects elements where both conditions are true.	<pre>[{"field1": "value", "field2": 15}, {"field1": "value", "field2": 20}]</pre>	[{"field1": "value", "field2": 15}, {"field1": "value", "field2": 20}]
<code>[.[] select(.field1 == "value" or .field2 > 10)]</code>	Selects elements where at least one of the conditions is true.	<pre>[{"field1": "value", "field2": 5}, {"field1": "other_value", "field2": 15}]</pre>	[{"field1": "value", "field2": 5}, {"field1": "other_value", "field2": 15}]
<code>[.[] select(.field1 == "value" and .field2 < 10)]</code>	Selects elements where both conditions are true.	<pre>[{"field1": "value", "field2": 5}, {"field1": "other_value", "field2": 15}]</pre>	[{"field1": "value", "field2": 5}]

JQ Filter	Description	JSON Sample	Result
[.[] select(.field1 == "value" and .field2 > 10 or .field3 == "include"))]	Combines and or conditions to filter elements where the first condition is true and either the second condition is true or the third condition is include.	[{"field1": "value", "field2": 5, "field3": "include"}]	[{"field1": "value", "field2": 5, "field3": "include"}]
[.[] select(.field1 != "value")]	Selects elements where the field1 is not equal to value.	[{"field1": "other_value", "field2": 15}, {"field1": "value", "field2": 5}]	[{"field1": "other_value", "field2": 15}]
[.[] select(.field1 == "value" and .field2 > 10 and .field3 == "include"))]	Combines and conditions to filter elements where the first condition is true and both the second and third conditions are true.	[{"field1": "value", "field2": 15, "field3": "include"}]	[{"field1": "value", "field2": 15, "field3": "include"}]
[.[] select(.field1 == "value" or .field2 > 10 and .field3 == "include"))]	Combines or conditions to filter elements where either the first condition is true or both the second and third conditions are true.	[{"field1": "other_value", "field2": 15, "field3": "include"}, {"field1": "value", "field2": 5, "field3": "include"}]	[{"field1": "other_value", "field2": 15, "field3": "include"}, {"field1": "value", "field2": 5, "field3": "include"}]
[.[] select(.field1 == "value" and .field2 > 10 and .field3 != "exclude")]	Combines multiple and conditions to filter elements where all conditions are true.	[{"field1": "value", "field2": 15, "field3": "include"}]	[{"field1": "value", "field2": 15, "field3": "include"}]

JQ Filter	Description	JSON Sample	Result
[.[] select(.field1 != = "value" and .field2 <= 10 or .field3 == "exclude"))]	Combines not and or conditions to filter elements where field1 is not equal to value and either field2 is less than or equal to 10 or field3 is exclude.	[{"field1": "other_value", "field2": 5, "field3": "exclude"}]	[{"field1": "other_value", "field2": 5, "field3": "exclude"}]
.[] .foo>30 and .bar== "less interesting data" .==true	Returns exactly true while executing rather than returning the values.	[{"foo": 42, "bar": "less interesting data"}]	true

Map Data and Create Lookups

A visual mapper enables you to map fields between applications with different data structures by dragging source fields onto target fields. Lookups associate 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 different vocabularies or systems.

Topics:

- [Map Data](#)
- [Manage Lookups](#)

Map Data

Use the mapper to map data between the source data structure and the target data structure.

Topics:

- [Access the Mapper](#)
- [Encode and Decode File Attachment Content](#)
- [Import Map Files](#)

See Map Data of *Using the Oracle Mapper with Oracle Integration 3*.

Access the Mapper

To create mappings in an integration, you must first access the mapper.

See Access the Mapper in *Using the Oracle Mapper with Oracle Integration 3*.

Encode and Decode File Attachment Content

The virtual file system (VFS) enables you to store files and internally use references to these files in the message payload. You can also map the VFS file's content to a string element.

For example, you can store files and use references in the VFS as follows:

- The REST Adapter supports the multipart attachment and application/octet-stream features. The attachment is stored in a staging area and an `attachmentReference` (string key) is generated. The `attachmentReference` key is sent as part of the message payload and later fetches the attachment instance from the staging area.
- The FTP Adapter uses `fileReference` for reading/writing a file without a schema. `fileReference` is also a reference to a file stored in the VFS.

In addition, mapping the VFS file's content to a string element enables you to:

- Map the content of a staged file attachment to a string element by converting the content to a base64 string.
- Store the base64 string as an attachment and generate a VFS reference.

Two XPath functions are provided to perform these tasks. These functions work with any adapter.

- `encodeReferenceToBase64(String reference)`: Accepts the VFS's file reference as input and returns the base64-encoded content of the file as the return value. This function has a file size limit of 10 MB. If a file is larger than 10 MB, an exception message of Maximum file size supported is 10 MB is displayed.
- `decodeBase64ToReference(String base64String)`: Accepts the base64-encoded content as input, decodes it, stores the base64-decoded value in a file in the VFS, and returns the reference to this file. There is no size limit because the content is already in memory.

The `77u/` text at the start of an encoded base64 string is related to the byte order mark and indicates the encoding scheme used (UTF-8). This is not needed for decoding of a base64 string to a binary.

This is needed for representing binary data in a string. This text must be extracted and removed from the original base64 string to ensure it doesn't cause decoding issues. If this is not removed from the encoded string, the decoded data, if converted to a string, has additional space.

You can extract and remove this from the base64-encoded string with the help of standard mapper functions for strings. This information is not needed to decode the base64-encoded string to binary data. This is needed for reading the binary data using UTF-8 encoding.

The location is the relative path (reference) of the file stored in Oracle Integration. The relative path is one of the following elements:

- `fileReference`
- `attachmentReference`
- `streamReference`

The two XPath functions are available for use in Oracle Integration:

- Expression Builder, when configuring the following actions in an integration:
 - Notification
 - Logging
 - Switch
 - Assign
- Mapper (visible after selecting  > Functions > Advanced)

The attachments are not restricted to document file types. For example, an image can be base64-encoded and later decoded back to the original file.

When an attachment is stored in the VFS, a key is generated to retrieve the attachment at a later time. The key is shown in the mapper as `attachmentReference/fileReference/streamReference`. This key is propagated within Oracle Integration as part of the payload. The attachment is claimed only when needed. The names `attachmentReference`, `fileReference`, and `streamReference` are based on the adapter type. For example, in the REST Adapter, `streamReference` is used. The data type of the reference is a string.

With a multipart feature, the HTTP request payload has multiple parts separated by boundaries. Each of the individual parts are considered an attachment. For raw bytes, `streamReference` is used. FTP uses `fileReference`.

Sometimes the endpoints accept only base64-encoded values. In these cases, the reference is passed as input to `encodeReferenceToBase64` to get the base64-encoded content of the file. Again, the base64-encoded value can be passed as input to `decodeBase64ToReference` to get the reference (location) to a file that contains the decoded content.

Import Map Files

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

Note:

As an alternative, you can directly edit XSLT code in the mapper. This eliminates the need to export your XSLT code from Oracle Integration, edit the code manually in a text editor or in a separate graphical tool such as Eclipse or Oracle JDeveloper, and then re-import the code into the mapper in Oracle Integration. See [Edit the XSLT Code in the Mapper](#) in *Using the Oracle Mapper with Oracle Integration 3*.

Topics:

- [Import a Map File into Oracle JDeveloper](#)
- [Import a Map File into Oracle Integration](#)

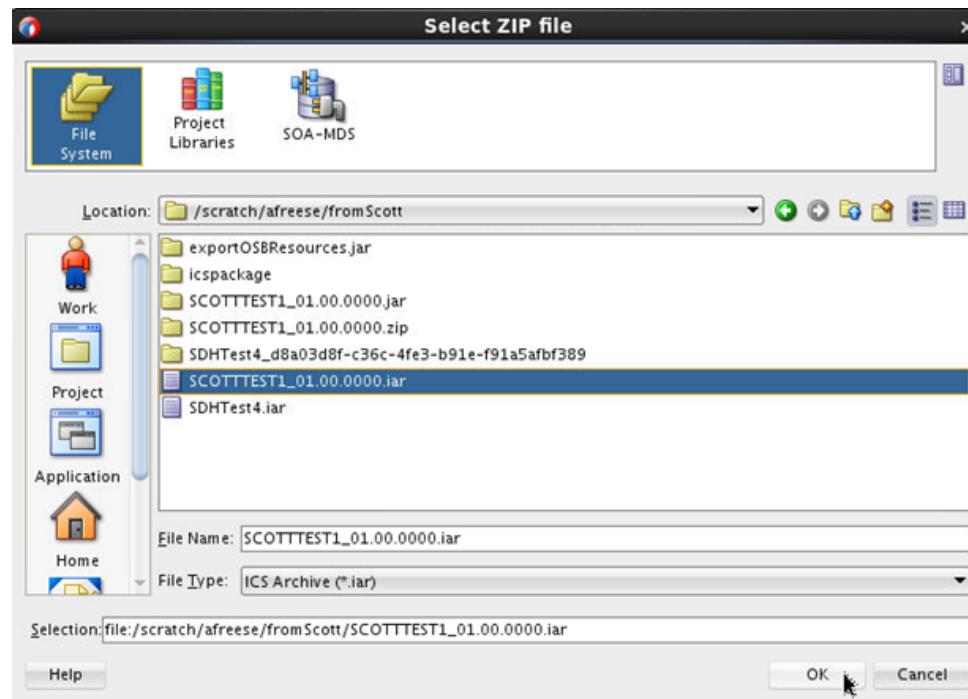
You can export an integration that includes a map file that you want to edit in Oracle JDeveloper. See [Export an Integration](#).

Import a Map File into Oracle JDeveloper

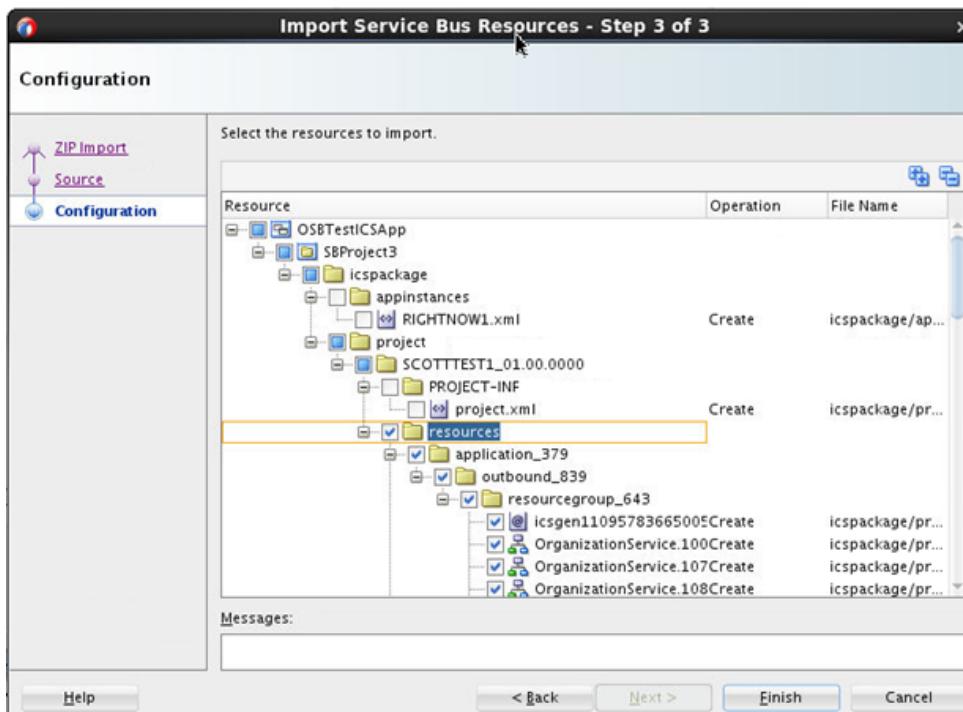
You can import an Oracle Integration archive file into an Oracle Service Bus project in Oracle JDeveloper. The archive file can include a map file that is largely complete in content or a map file that is empty of content. This action enables you to perform advanced XSLT tasks (create variables, use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

1. See [Export 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.x, perform the following steps:
 - a. From the **File Type** menu, select **ICS Archive (*.iar)**.

- b. Browse for and select the Oracle Integration 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.

Import a Map File into Oracle Integration

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 Oracle Integration 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 Oracle Integration. The map file must be from an Oracle Service Project in Oracle JDeveloper.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Click the specific integration in which to import the map file.
3. Click the mapper icon, then select **...**, then **Import**.
4. 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 Oracle Integration. You only import the map file of the exported integration back into Oracle Integration.
5. Click **Import**.

Manage Lookups

A lookup associates values used by one application for a specific field with the values used by other applications for the same field. This capability enables you to map values across

vocabularies or systems. For example, you can map country codes, city codes, currency codes, and so on.

 **Note:**

Specifying a lookup value that includes a special character of # is *not* supported.

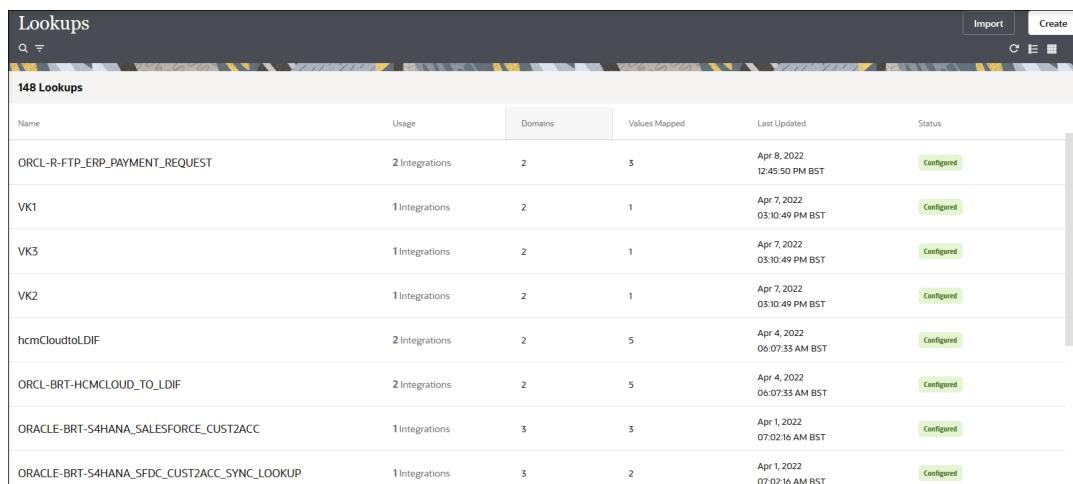
Topics:

- [Create a Lookup](#)
- [Clone a Lookup](#)
- [Delete a Lookup](#)
- [Update Lookups in Active Integrations](#)
- [Export a Lookup](#)
- [Import a Lookup](#)
- [Create the lookupValue Function](#)

Create a Lookup

Create a lookup to map values between applications.

1. In the navigation pane, click **Design**, then **Lookups**.



Name	Usage	Domains	Values Mapped	Last Updated	Status
ORCL-R-FTP_PAYMENT_REQUEST	2 Integrations	2	3	Apr 8, 2022 12:45:50 PM BST	Configured
VK1	1 Integrations	2	1	Apr 7, 2022 03:10:49 PM BST	Configured
VK3	1 Integrations	2	1	Apr 7, 2022 03:10:49 PM BST	Configured
VK2	1 Integrations	2	1	Apr 7, 2022 03:10:49 PM BST	Configured
hcmCloudtoLDIF	2 Integrations	2	5	Apr 4, 2022 06:07:33 AM BST	Configured
ORCL-BRT-HCMCLOUD_TO_LDIF	2 Integrations	2	5	Apr 4, 2022 06:07:33 AM BST	Configured
ORACLE-BRT-S4HANA_SALESFORCE_CUST2ACC	1 Integrations	3	3	Apr 1, 2022 07:02:16 AM BST	Configured
ORACLE-BRT-S4HANA_SFDC_CUST2ACC_SYNC_LOOKUP	1 Integrations	3	2	Apr 1, 2022 07:02:16 AM BST	Configured

2. Click **Create**.

The Create lookup panel opens.

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

Add Adapters or Domain Names to a Lookup

Add adapters or domain names to a lookup to map values between connections.

1. Click **Domain_Name_1**.

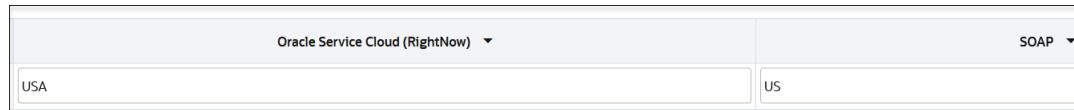
A menu with options is displayed.

2. Select the type to create:

- **Replace with adapter:** This selection invokes the Select adapter dialog for choosing the adapter to use. You can change your selection:
 - If you want to change your adapter selection, click the selected adapter type, then select a different adapter.
 - If you want to change from an adapter to a domain name, click the selected adapter type and change to a domain name.
- **Edit domain name:** This selection invokes a dialog for entering the domain name to use. There is a 50-character limit. You can change your selection:
 - If you want to change from a domain name to an adapter, click the domain name, then select **Replace with adapter**.
 - If you want to change the domain name, click the name, then select **Edit domain name**.

The adapter or domain name is displayed.

- 3.** Enter a value in the field after the adapter or domain name. To add more rows for additional values, click the + sign.
- 4.** Click **Domain_Name_2**.
- 5.** Repeat Step 2 to add an adapter or domain name.
- 6.** Enter a value in the field after the adapter or domain name. To add more rows for additional values, click the + sign.



USA	US

- 7.** If you want to add another column, click +.
- 8.** Repeat to add more adapters and domain names to the lookup.
- 9.** Click **Save** when complete.

Clone a Lookup

You can clone a copy of an existing lookup in Oracle Integration. Cloning provides a quick way to create a new lookup with similar information. Lookups with adapters and domain names can be cloned.

- 1.** In the navigation pane, click **Design**, then **Lookups**.
- 2.** Hover over the lookup to clone.
- 3.** Click **Actions ...**, then select **Clone**.
The Clone lookup panel opens.
- 4.** Enter a name and an optional keyword and description.
- 5.** Click **Clone**.

Delete a Lookup

You can delete lookups in Oracle Integration that are not currently in use by integrations.

1. In the navigation pane, click **Design**, then **Lookups**.
2. Hover over the lookup to delete.
3. Click **Actions ...**, then select **Delete**.

The Delete lookup dialog is displayed.

You cannot delete a lookup currently used by an integration. Deleting the lookup would invalidate the integration and cause an active integration to stop running. You must first remove the lookup from the integration and then delete the lookup.

4. Click **Delete** when prompted to confirm.

Update Lookups in Active Integrations

Lookups are not deployed as part of integration activation. Therefore, changes that you make to lookups already used in active integrations typically take effect immediately. There is no need to re-activate integrations using a changed lookup for the new lookup value to take effect.

There can be instances in which updates to the runtime environment are slow because it is not in real-time and queuing can occur at runtime. For those cases, updates can occur in a couple of minutes.

Export a Lookup

Once you create a lookup, you can export that lookup for use in other Oracle Integration environments.

1. In the navigation pane, click **Design**, then **Lookups**.
2. Hover over the lookup to export.
3. Click **Actions ...**, then select **Export to CSV**.

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

Import a Lookup

You can import lookups that were previously exported from Oracle Integration.

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 Oracle Integration. This name cannot contain spaces.

DVM, *table_name*

1. Locate the CSV file containing the lookup table you want to import.
2. In the navigation pane, click **Design**, then **Lookups**.
3. In the banner, click **Import**.

The Import lookup panel opens.

4. Click **Drag and Drop** to select a lookup or drag one to this field.

5. 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. You can customize or activate it, if it is ready.

Create the `lookupValue` Function

You can create the parameter values for the `lookupValue` function with the Map Lookup Value Wizard. This wizard enables you to define the lookup table, source column, target column, and default value to use in the function. For these parameter values to be selectable in the wizard, you must have already created a lookup on the Lookups page.

Topics:

- [Access the Build Lookup Function Wizard](#)
- [Select the Lookup](#)
- [Select the Source and Target Columns](#)
- [Specify the Default Value](#)
- [Review Your Lookup Table Selections](#)

Access the Build Lookup Function Wizard

The Map Lookup Value Wizard for creating the `lookupValue` function parameter values is accessible from the mapper or the Expression Builder in actions that support functions in Oracle Integration.

S

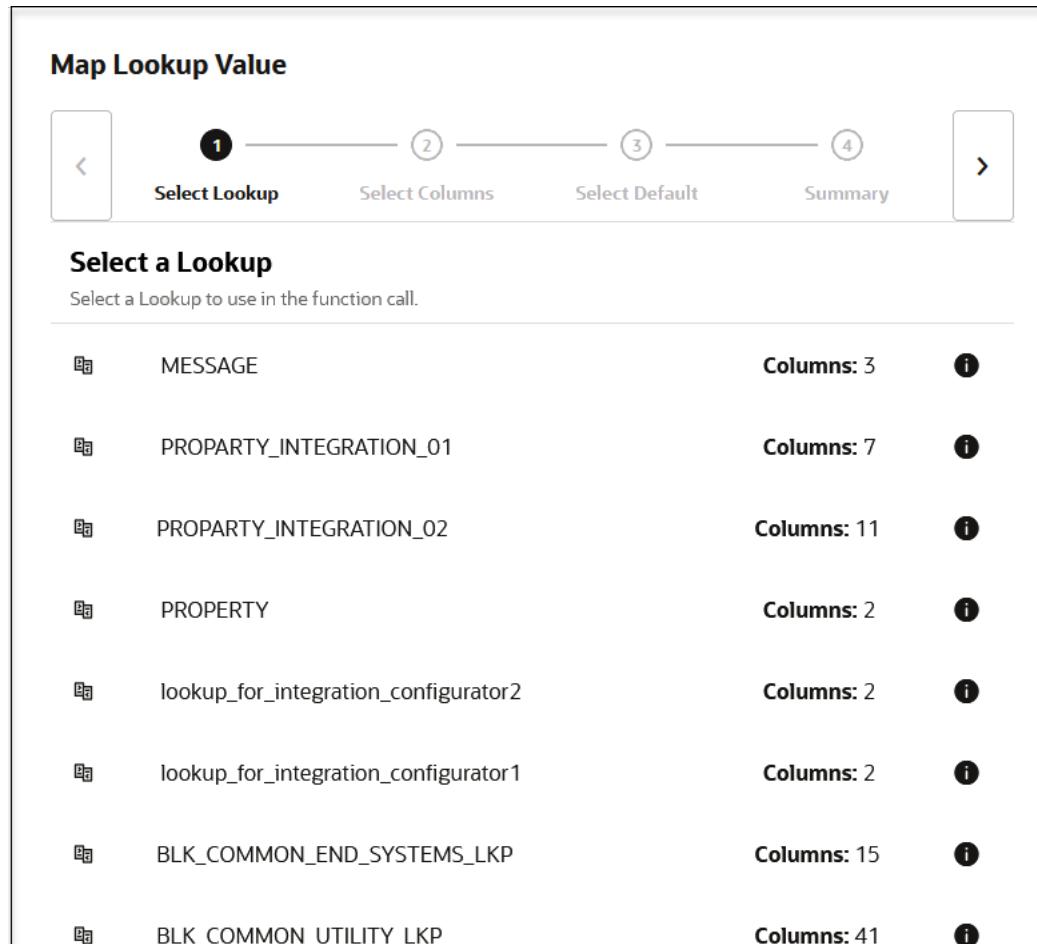
Note:

You must already have created lookups to use this wizard.

- [From the Mapper](#)
- [From the Expression Builder In an Action that Supports Functions](#)

From the Mapper

1. In the upper right corner, click **Toggle functions**  to launch the **Components** panel.
2. Type `lookupValue` in the **Search** field, and click **Search**.
3. Drag the function onto the target element node.
The Map Lookup Value Wizard is displayed.



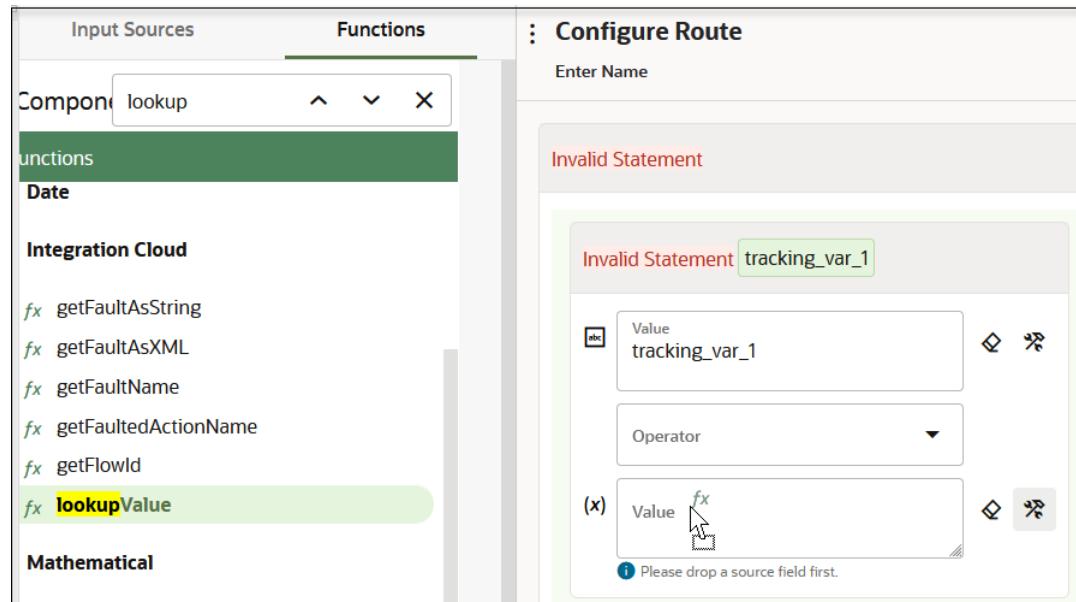
From the Expression Builder in an Action that Supports Functions

You can access the Map Lookup Value Wizard in the Expression Builder in an action that supports functions, such as a switch, stitch, or others.

1. Open an integration for editing.
 2. Open an action that supports functions in the integration (for this example, a switch action is shown, but other actions such as a stitch can be used).
- The Configure route panel opens.
3. Manually enter a value, select a value from the drop-down list, or drag a value from the **Sources** tree to the first **Value** field. For this example, **tracking_var_1** is selected.
 4. Select an operator in the **Operator** field.
 5. Click **Functions**, and browse or search for `lookupValue`.
 6. Click `lookupValue`.
 7. View the description after the function for information about the supported parameters. You define values for these parameters in the Map Lookup Value Wizard.
 - `dvm-location`
 - `src-column`
 - `src-value`

- target-column
- default-value

8. Click **Switch to Developer View**  in the second **Value** field.
9. Drag the `lookupValue` function into the second **Value** field.



The Map Lookup Value Wizard is displayed.

Map Lookup Value

1 Select Lookup 2 Select Columns 3 Select Default 4 Summary

Select a Lookup

Select a Lookup to use in the function call.

MESSAGE	Columns: 3	
PROPROPERTY_INTEGRATION_01	Columns: 7	
PROPROPERTY_INTEGRATION_02	Columns: 11	
PROPERTY	Columns: 2	
lookup_for_integration_configurator2	Columns: 2	
lookup_for_integration_configurator1	Columns: 2	
BLK_COMMON_END_SYSTEMS_LKP	Columns: 15	
BLK_COMMON.Utility_LKP	Columns: 41	

Create the `lookupValue` function parameter values.

Select the Lookup

You can select the lookup to use in the `lookupValue` function. You must already have created a lookup. Otherwise, no lookups are displayed for selection.

1. Select the lookup to use in the function. You can view the lookup description by clicking the information icon in the row. This can guide you in selecting the required lookup. The number of columns defined in the lookup is also displayed.
2. Click **Next >**.

Select the Source and Target Columns

You can select the source and target columns to use in the `lookupValue` function. The `lookupValue` function requires one source column and one target column. When you select the source and target columns, the values available with the columns are displayed.

1. Select the source and target columns to use in the `lookupValue` function.

Element	Description
Select Source	Click the source column header to select from a list of available columns for this lookup table. The data included with the selected column is displayed. Both adapter and domain name columns are displayed.
Select Target	Click the target column header to select from a list of available columns for this lookup table. The data included with the selected column is displayed. Both adapter and domain name columns are displayed.

2. Click **Next >**.

Specify the Default Value

You must specify a default value to use.

1. Enter the default value to use in the `lookupValue` function if no match is found. If there is no match that satisfies all the search values, the lookup fails and the default value is returned (for example, an actual default value to use or an error message such as `No Value Found`).
2. Click **Next >**.

Review Your Lookup Table Selections

You can review the lookup table values to use in the `lookupValue` function on the Summary page.

1. Review the lookup table values. The Summary page is the final wizard page after you complete your configuration.

Element	Description
Parameter and Value Table	<p>Displays a summary of the parameters and values you defined on previous pages of the wizard.</p> <p>To return to a previous page to update any values, click the appropriate tab in the left panel or click Back .</p>

Element	Description
Resulting Expression	<p>Displays the expression you defined on the previous pages of the wizard. The <code>lookupValue</code> function takes the following format:</p> <pre>lookupValue(dvm-location, src-column, src-value, target-column, default-value)</pre> <p>Where:</p> <ul style="list-style-type: none"> • Lookup Table: The lookup table selected on the Select lookup page. • Source Column: The source column selected on the Select columns page. • Source Value: The source value you enter in the mapper or Expression Builder after completing this wizard. Complete this wizard, then define the <code>srcValue</code> parameter value. • Target Column: The target column selected on the Select columns page. • Default Value: The default value entered on the Default value page. <p>For example, a defined <code>lookupValue</code> function after you have completed the wizard and defined the <code>srcValue</code> parameter value in the Expression Builder can look as follows:</p> <pre>dvm:lookupValue('countrycode', 'countrycode', srcValuePlaceHolder, 'name', 'No data found')</pre>

For example:

Map Lookup Value

1 Select Lookup 2 Select Columns 3 Select Default 4 Summary

Summary

Confirm the resulting expression. Note the 'Source Value' is a placeholder. You will need to select the 'Source Value' in the main editor after clicking on Done.

countrycode

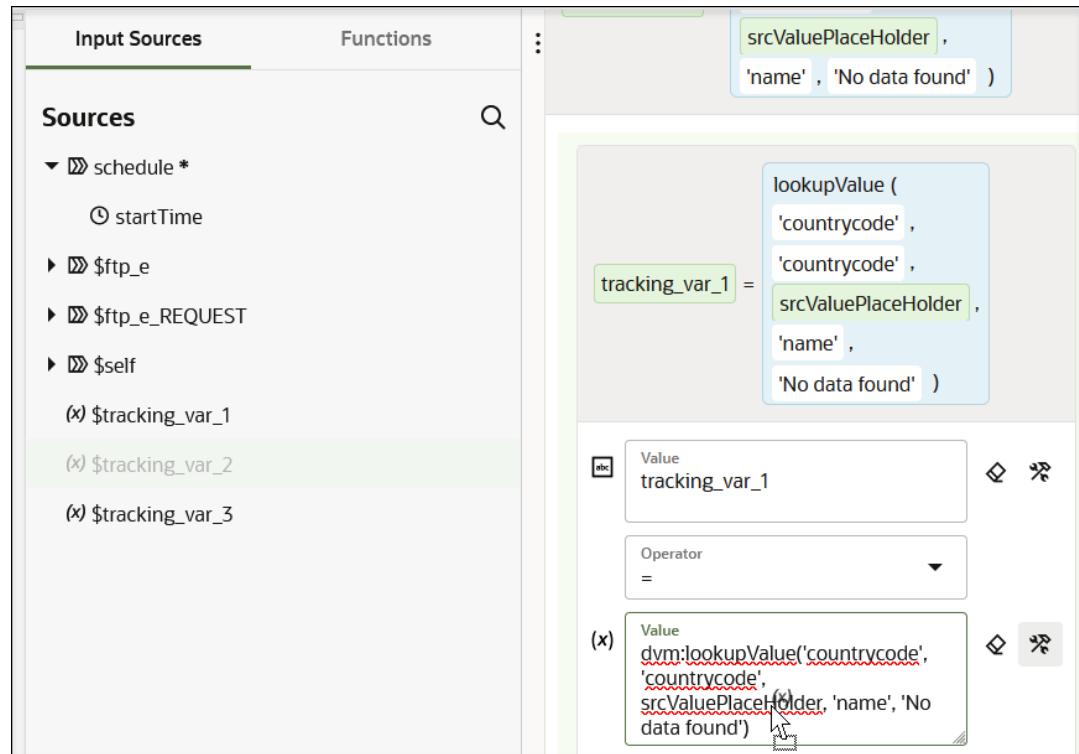
Parameter	Value
Lookup Table	countrycode✓
Source Column	countrycode✓
Source Value	Note: The value for this parameter should be set in the editor.
Target Column	name✓
Default Value	No data found✓

Resulting Expression:

```
dvm:lookupValue('countrycode', 'countrycode', srcValuePlaceHolder, 'name', 'No data found')
```

Cancel **Done**

2. Click **Done**.
3. In the **Sources** tree, drag a value to replace **srcValuePlaceHolder** in the **Value** field.



4. Click **Save.**

You can use the configured function in the mapper. See Add Functions, Operators, and XSLT Statements in *Using the Oracle Mapper with Oracle Integration 3*.

Manage the Agent Group and Connectivity Agent

You must create an agent group and install the on-premises connectivity agent before you can create an integration in which messages are exchanged between Oracle Integration and your locally-installed, application resources residing in an on-premises network or a private cloud.

Topics:

- [Workflow for Using the Connectivity Agent](#)
- [Requirements for the Connectivity Agent](#)
- [Key Points for the Installation of the Connectivity Agent](#)
- [Download and Run the Connectivity Agent Installer](#)
- [Restart the Connectivity Agent](#)
- [Create a Connection with an Agent Group](#)
- [Upgrade the Connectivity Agent Automatically](#)
- [Recover the Connectivity Agent in Case of Host Failure](#)
- [Use the Connectivity Agent in High Availability Environments](#)
- [Delete an Agent Group](#)
- [Agent Behavior in a Decommissioned Instance or HTTP 404/401 Error Response Codes](#)

Conceptual details and patterns for using the connectivity agent are provided. See [About Creating Hybrid Integrations](#).

If you experience problems with the on-premises connectivity agent, troubleshooting information is provided. See [Troubleshoot the Connectivity Agent](#).

Workflow for Using the Connectivity Agent

Follow this workflow to use the on-premises connectivity agent.

Task	Documentation
Create a connectivity agent group.	Create an Agent Group
Download and run the on-premises connectivity agent installer on your host.	Download and Run the Connectivity Agent Installer
During installation setup, you associate the on-premises connectivity agent with the agent group.	
Create an adapter connection in Oracle Integration and associate the connection with the connectivity agent group.	Create Connections
Design an integration that uses this connection.	Create an Integration
Activate the integration.	Activate an Integration

Requirements for the Connectivity Agent

If your organization requires one or more connectivity agents, make sure that the connectivity agent is compatible with your organization's operating procedures.

The connectivity agent has the following requirements:

- Your organization must allow outbound calls from your intranet. If your organization has a sensitive network, any outbound communication to outside the firewall might be restricted.

The connectivity agent must establish an outbound call from your intranet to Oracle Integration. Every interaction that the connectivity agent has with Oracle Integration follows a prescribed paradigm in which the connectivity agent pulls information from your Oracle Integration instance.

- You must make sure that your organization allows the connectivity agent to use OAuth 2.0 token-based authentication when invoking Oracle Integration endpoints.

See Create an Agent Group.

- You must make sure that the on-premises host, often a virtual machine, meets the system requirements.

See System Requirements.

- You must make sure that the functionality of the connectivity agent is compatible with your organization's procedures and requirements.

See About the Connectivity Agent.

- The person who installs the connectivity agent must have the required permissions and roles.

See Agent Download and Installation.

Key Points for the Installation of the Connectivity Agent

- The connectivity agent comes preseeded with the credentials that it needs to access your Oracle Integration instance.
- The connectivity agent must be able to access the application that you're integrating with. Therefore, the host that you install the connectivity agent on must also have access to the application.
- Install the connectivity agent to the instance with the minimal privilege. Do not install it as root.
- Make sure you install the connectivity agent with the privileges that it needs to run.
- After successfully installing the connectivity agent, delete the installation key.
- Verify that ports or proxy are set up correctly to only allow for the connectivity agent connection to Oracle Cloud Infrastructure.
- Configure the connectivity agent to scan only the specified directories at the specified frequency.

Download and Run the Connectivity Agent Installer

You must create an agent group and download and run the connectivity agent installer to install the agent in your local environment.

- [Create an Agent Group](#)
- [System Requirements](#)
- [Connectivity Agent Restrictions](#)
- [Agent Download and Installation](#)

Create an Agent Group

You must create an agent group in Oracle Integration before you can run the connectivity agent installer. Creating the agent group automatically creates an OAuth client application in your identity domain or Oracle Identity Cloud Service environment. This enables the connectivity agent to use OAuth 2.0 token-based authentication when invoking Oracle Integration endpoints. Basic authentication is *not* supported in Oracle Integration 3. Up to two connectivity agents can be associated with an agent group. For a single Oracle Integration instance, you can create up to five agent groups.

- [What to Know About OAuth 2.0 Token-Based Authentication Support](#)
- [Behavior When Two Connectivity Agents are Associated with an Agent Group](#)
- [Create an Agent Group](#)

What to Know About OAuth 2.0 Token-Based Authentication Support

OAuth 2.0 token-based authentication is more secure and Oracle Integration 3 connectivity agent configuration is simpler than in Oracle Integration Generation 2.

However, be aware of the following:

- You must allow egress from the agent network to Oracle Integration design-time and runtime and the identity domain or Oracle Identity Cloud Service.
- The Service Gateway on the Oracle VCN provides access to Oracle Integration, Oracle Identity Cloud Service, and identity domains in the same region as the connectivity agent. See [Access to Oracle Services: Service Gateway](#).
- The Service Gateway on the Oracle VCN does not provide access to Oracle Integration, Oracle Identity Cloud Service, and identity domains in a different region than the connectivity agent; this type of access requires the NAT Gateway. See [NAT Gateway](#).

Behavior When Two Connectivity Agents are Associated with an Agent Group

Up to two connectivity agents can be associated with an agent group. This recommended setup provides the following benefits:

- Makes the connectivity agent highly available
- Eliminates the potential for a single point of failure if you are using only a single connectivity agent that then becomes unavailable

When creating the agent group, you can enable the **Use a Single Agent at a Time** option to control connectivity agent behavior, as described in the following table. Note that running adapters with **Use a Single Agent at a Time** enabled can result in lower performance when compared to agent groups in which **Use a Single Agent at a Time** is not enabled.

If You ...	Behavior of the Two Agents	Which Adapters Can Use This Behavior
Enable the Use a Single Agent at a Time Option	<p>Only one connectivity agent is active at all times. The other connectivity agent is in passive (standby) mode.</p> <ul style="list-style-type: none"> The agent started/installed first is initially identified as primary and active. The second agent is identified as passive and placed in standby mode. The active agent handles all incoming requests and polling responsibilities (for example, a File Adapter polling for files in a specific directory or a database adapter polling for new inserts). The passive agent remains in standby mode and does not handle any incoming requests or polling responsibilities. <p>If the active agent fails, the passive agent becomes active and handles all incoming requests and polling responsibilities. This agent remains active even after the initially active agent recovers from failure. The recovered agent is now passive and remains in standby mode.</p>	<ul style="list-style-type: none"> If you use the MLLP Adapter, you must select an agent group that has enabled Use a Single Agent at a Time. The .MLLP Adapter can only work with one agent active at a time. If you use the File Adapter or FTP Adapter, it is recommended that you select an agent group that has enabled Use a Single Agent at a Time. While the File Adapter or FTP Adapter works without Use a Single Agent at a Time enabled, this option limits resiliency during agent failure. With Use a Single Agent at a Time enabled, you can install two agents. The second agent runs in standby mode, ready to take over if the primary agent fails. Ensure that the directory paths to poll/ read/ write files are accessible to both agents in the agent group. See File Adapter Capabilities in <i>Using the File Adapter with Oracle Integration 3</i> and FTP Adapter Capabilities in <i>Using the FTP Adapter with Oracle Integration 3</i>. If you use any other adapter, this option is not required. Select this option based on your business requirements.
Don't enable the Use a Single Agent at a Time Option	Both agents remain active at all times. Incoming requests and polling responsibilities are shared by both agents.	<ul style="list-style-type: none"> The MLLP Adapter <i>cannot</i> select an agent group that has not enabled this option. Only agent groups with the Use a Single Agent at a Time option enabled (identified by the Singleton label) can be selected. All other adapters can select an agent group that has not enabled this option.

Create an Agent Group

Creating the agent group also creates the necessary artifacts required for message exchange. You can edit the connection name after creation.

- In the navigation pane, click **Design**, then **Agents**.
 - Click **Create**.
- The Create agent group panel opens.
- Enter the following information, then click **Create**.

Field	Description
Name	<p>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:</p> <ul style="list-style-type: none"> • Letters (A-Z, a-z) • Numbers (0-9) • Spaces () • Special characters (_ -) <p>The name must not begin or end with a space and cannot be longer than 50 characters.</p>
Identifier	<p>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 agent, the identifier value is automatically included in the <code>InstallerProfile.cfg</code> file.</p> <p>Note: After creating the agent group, you cannot edit the agent group identifier. Instead, you must delete and recreate another agent group to associate with a different agent group identifier.</p>
Description	<p>Provide a meaningful description so that others can understand the responsibilities of the agent group.</p>
Use only one agent at a time	<p>Enable this option if you want only a single connectivity agent active at a time to handle all incoming requests and polling responsibilities. Associate two connectivity agents with this agent group. Then associate this agent group with your adapter connection in the Access type section of the Connections page. See Behavior When Two Connectivity Agents are Associated with an Agent Group.</p> <p>If the Use only one agent at a time option is enabled, the label Singleton appears next to the agent group name after creation.</p> <p>Note:</p> <ul style="list-style-type: none"> • Once Use only one agent at a time is enabled, it cannot be disabled. You must delete and then recreate the agent group. • If you select Use only one agent at a time and then associate only one agent with the agent group, it works as if that option were never selected. You must associate a second agent with that agent group to achieve active-passive agent functionality.

This action automatically creates an OAuth client application in the identity domain or Oracle Identity Cloud Service. The application name takes the following form:

`agent_group_name-instance_identity_provider_name`

This enables the connectivity agent to use OAuth 2.0 token-based authentication when invoking endpoints. In Oracle Integration 3, each agent group has its own OAuth client application.

System Requirements

Satisfy the following system requirements before installing the on-premises connectivity agent.

- [Connectivity Agent Prerequisites](#)
- [Ensure the Connectivity Agent Version is Compatible with JDK 17](#)
- [Upgrade the Existing JDK 11 Agent to the JDK 17 Agent](#)

Connectivity Agent Prerequisites

- Install and use JDK version 17.

You do not need to purchase a JDK license for the sole purpose of installing and/or running the connectivity agent. See [JDK license and support for OIC Connectivity Agent \(Doc ID 2611142.1\)](#).

The JDK installation can be the same one as used with other products installed on the same host. However, separate JDK licensing may be required for any use of Java beyond the use of the connectivity agent. Ensure that the JDK installation is *not* modified for use with these other products.

- Install the connectivity agent on one of the following certified operating systems:
 - Oracle Linux 6.x
 - Oracle Linux 7.x
 - Oracle Linux 8.x
 - RedHat Enterprise Linux 6.6
 - RedHat Enterprise Linux 7.x
 - RedHat Enterprise Linux 8.x
 - Suse Linux Enterprise Edition 12 SP2
 - Windows Standard Edition 2016
 - Windows 2019

 **Note:**

- The connectivity agent is not certified on Kubernetes (K8s).
- IBM or Open JDK are *not* supported.

- Allowlist the Oracle Integration runtime and design-time IP addresses and the Oracle Identity Cloud Service IP address to install the connectivity agent. This prerequisite is *only* required if you restrict access to the internet based on the target IP address.
- Provide a minimum of 8 GB memory with 4 GB of heap size dedicated for the connectivity 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 a value greater than 8 GB.

Ensure the Connectivity Agent Version is Compatible with JDK 17

Run the following command from the `agenthome` directory to determine if the connectivity agent is compatible with JDK 17. The following minimum version on Oracle Integration 3 is required for JDK 17 compatibility.

```
$ cat version
20230613.0926.318
```

If you receive the following error when starting the connectivity agent after upgrading to JDK 17, you are not running a compatible version of the connectivity agent:

```
Agent is only supported on Java version 8 or 11. Please set JAVA_HOME and
PATH to the location
for Java (JDK 8 or 11). This error occurs because you are using an older
version of the connectivity agent.
```

You may be running an older version of the connectivity agent for the following reasons:

- Previous attempts to upgrade the connectivity agent failed
- You started the agent with the following option:

```
-DenableAutoUpgrade=false
```

To resolve this issue:

1. Start the connectivity agent once with your previous JDK version (JDK 8 or JDK 11).
2. Wait for the connectivity agent upgrade to finish. You can monitor upgrade in the `agenthome/logs/agent-upgrade.log` file.
3. Stop the connectivity agent.
4. Restart the connectivity agent with JDK 17.

Upgrade the Existing JDK 11 Agent to the JDK 17 Agent

If you need to upgrade from JDK 11, perform the following steps.

1. Install JDK 17.
2. Stop the existing connectivity agent.
3. Set `JAVA_HOME` and `PATH` to JDK17.
4. Restart the connectivity agent.

Connectivity Agent Restrictions

Note the following connectivity agent restrictions.

- When the Oracle WebLogic JMS Adapter is used with a version of Oracle WebLogic Server prior to 12.2.1.4, JDK 11 and JDK 17 are not supported. To know your version of Oracle WebLogic Server, perform the following steps:
 1. Enter the Oracle WebLogic Server URL. For example:

```
http://abcweblogic.oracle.com:7001/console
```
 2. At the bottom left, find the version. For example:

```
WebLogic Server Version: 12.2.1.4.0
```

- When installed on a Windows host, the connectivity agent does not work with a proxy using NT LAN Manager (NTLM) authentication.

Agent Download and Installation

The agent must be downloaded and installed on your local host. If needed, you can also install a security certificate on your local host. If necessary, you can also run the agent installer as a background process.

- [Download and Install the Agent](#)
- [Install a Certificate on the Agent Host](#)
- [Run the Connectivity Agent Installer as a Background Process on Linux Systems](#)

Download and Install the Agent

Download the connectivity agent installer to install the agent in your local environment. During installation, you associate the connectivity agent with the agent group identifier you generated when creating an agent group in Oracle Integration. Ensure that you restrict access to the folder in which you install the connectivity agent to only those users who need to stop and start the agent.

 **Note:**

- You must have the Java execute permission to install and restart the agent.
- You must have the ServiceAdministrator role to download the agent.
See [What Users Can Do in the Integrations Design Section by Role of Provisioning and Administering Oracle Integration 3](#).

1. Create a directory for connectivity agent installation on your on-premises host.

 **Note:**

- Do not install the agent in a directory path that includes `/tmp`.
The agent must never be installed in `/tmp`. Per the Filesystem Hierarchy Standard version 3.0, `/tmp` is meant for temporary files. Even though the install and agent work, it is not a recommended location for agent installation because the directory in `/tmp` may be deleted after the reboot of the system or agent virtual machine.
- Agent installation is not supported with use of an SSL proxy.

2. In the navigation pane, click **Design**, then **Agents**.
3. Click **Download**, then **Connectivity Agent**.
4. Download the connectivity agent installer to a directory on your on-premises host.
5. Extract `oic_conn_installer.zip`.
6. If you need to add any third party JARs (for example, for the Siebel Adapter or MySQL Adapter), copy them under the `agenthome/thirdparty/lib` directory.

 **Note:**

If you perform this step after installing the connectivity agent, you must restart the agent. See [Restart the Connectivity Agent](#).

7. Hover over the agent group.

8. Click **Actions . . .**, then select **Download config**.

This step downloads a preconfigured `InstallerProfile.cfg` file to use during installation. The following message is displayed.

BootStrap configuration downloaded for Agent Group `agent_group_name`. Replace the file in agent install location before proceeding with agent setup.

9. Replace the existing `InstallerProfile.cfg` template file in the `oic_conn_agent_installer` directory that was created when you extracted the agent installation file in Step 5 with the preconfigured `InstallerProfile.cfg` file you downloaded in Step 7.

The preconfigured `InstallerProfile.cfg` file automatically includes values for all required parameters such as `oic_URL` and `agent_GROUP_IDENTIFIER` and OAuth 2.0 token-based authentication parameters such as client ID, client secret, and scope. This eliminates the need to manually specify values for these parameters.

- a. Optionally set proxy parameter values. These parameters are only required if the connectivity agent is used with a proxy in the on-premises environment.
- b. If you have multiple hosts that must be configured in a nonproxy host environment, you must separate each IP address or host with a pipe symbol (|) in the `proxy_NON_PROXY_HOSTS` parameter. For example:

```
proxy_NON_PROXY_HOSTS=localhost|127.0.0.1|*.myorg.com
```

- c. If you use a proxy user for Windows, the user name must include the MS domain name in front of the user name, along with double backslashes before the user name (for example, `MS_domain\\username`). If you do not specify the double backslashes, you receive a 407 Proxy Authentication Required error.

10. Set the `JAVA_HOME` property to the location of the JDK installation.

11. Set the `PATH` property. For example, if `csh` is your shell:

```
setenv PATH = $JAVA_HOME/bin:$PATH
```

12. Run the connectivity agent installer from the command prompt. If you want to copy and paste this command, ensure that it does not have any special characters.

```
java -jar connectivityagent.jar
```

13. Wait for a successful installation message to appear.

Done with Agent installation & configuration... Starting agent for message processing.

Agent started successfully... listening for new messages...

The agent group associated with this connectivity agent now has its own `InstallerProfile.cfg` file and OAuth client application in Oracle Identity Cloud Service. If you delete the agent group, the OAuth client application is also deleted in Oracle Identity Cloud Service.

If errors occur, review the agent diagnostic logs.

See [Monitor Agents](#).

14. Depending on your agent environment, you may also need to install a certificate.

See [Install a Certificate on the Agent Host](#).

Install a Certificate on the Agent Host

If you need to add a certificate on the agent host, use the `keytool` to import the certificate in `keystore.p12`. Installing the certificate enables you to access hosts with self-signed certificates. A certificate is not typically needed.

Note:

If you install a certificate *after* installing the connectivity agent, you must restart the agent.

See [Restart the Connectivity Agent](#).

Scenarios under which you need to import the certificate in the agent keystore are as follows:

- The connectivity agent is used with an SSL proxy.
 - The connectivity agent is used to invoke secure (SSL) on-premises endpoints.
1. Go to the `agenthome/agent/cert/` directory (`keystore.p12` is available here).
 2. Run the following command:

```
keytool -importcert -keystore keystore.p12 -storepass password -alias alias_name -noprompt -file certificate_file -trustcacerts
```

Where:

- `-storepass password`: The default, initial password for the agent keystore. Refer to your `keytool` documentation for the default `storepass` password. See [keytool](#).
- `-alias alias_name`: Any name to uniquely identify the imported certificate in the keystore.
- `-file certificate_file`: Absolute path of the certificate file.

Run the Connectivity Agent Installer as a Background Process on Linux Systems

When you run the connectivity agent installer (using `java -jar connectivityagent.jar`), the process is tied to the terminal window in which you are working and ends when the window is closed. If you want to run the process in the background, use one of the following options:

- [Run the Connectivity Agent Installer as a Background Process](#)
- [Connectivity Agent Cannot Be Run as a Service on Windows](#)

Run the Connectivity Agent Installer as a Background Process

1. Run the agent process. For example:

```
nohup java -jar connectivityagent.jar &
```

Or

1. If you do not want to expose the password in `InstallerProfile.cfg`, perform the following steps:
 - a. Enter `java -jar connectivityagent.jar` at the command prompt.
 - b. Enter the user name and password when prompted.
 - c. Enter `Ctrl+Z` to suspend the process.
 - d. Enter `bg` (to run the process in the background).
 - e. Enter `jobs` to get the jobid.
 - f. Enter `disown -a %jobid` (from Step e) to disassociate the process from the owning shell.

Connectivity Agent Cannot Be Run as a Service on Windows

Running the connectivity agent as a service on Windows is *not* supported. You must be logged in to the Windows host. Otherwise, the agent stops working when you log off. Consider the following recommended options.

1. Create a new service account on Windows.
2. Use that account to log in to the Windows host and remain logged in with that account.
3. Install the connectivity agent on the Windows host.
4. Use that agent to communicate with the MS SQL Server or other endpoints accessed on Windows.

Or

1. Instead of installing the connectivity agent on a Windows host, install it on a Linux virtual machine that has access to the Windows systems with which that agent must interact.

Restart the Connectivity Agent

You can restart the on-premises connectivity agent if required.

1. Stop the agent in either of the following ways:
 - Enter `ctrl+c` on the host on which the agent is running.
 - Search for the connectivity agent process and terminate it.
2. Wait at least 45 seconds before restarting the agent. This is because the agent monitoring framework waits for 45 seconds before marking the agent status as being down. If you start the agent before 45 seconds have completed, agent restart fails with the following error message:

Agent is already running for this particular instance

3. Restart the agent based on your environment and production load size.

Environment	Enter the Following Command
Production environments	<p>It is recommended that you restart the agent with the <code>-XX:+HeapDumpOnOutOfMemoryError</code> parameter:</p> <pre>java -XX:+HeapDumpOnOutOfMemoryError -jar connectivityagent.jar</pre> <p>If the connectivity agent runs out of memory, this parameter by default ensures that the heap dump is stored in a <code>java_pidpid.hprof</code> file in the directory where the agent application is run. Based on production loads, it may sometimes be necessary to allocate a larger amount of heap size for the agent process. If you determine that the process must be allocated a larger heap size, tune the <code>-Xms</code> and <code>-Xmx</code> parameters accordingly.</p> <ul style="list-style-type: none"> • <code>-Xmsheap_sizeG</code> • <code>-Xmxheap_sizeG</code> <p>For example, assume you want to assign a minimum of 2 GB and a maximum of 8 GB to the agent JVM.</p> <pre>-Xms2G -Xmx8G</pre> <p>Start the agent with the following parameters.</p> <pre>java -Xms2G -Xmx8G -jar connectivityagent.jar</pre> <p>You can specify the <code>-XX</code>, <code>-Xms</code>, and <code>-Xmx</code> parameters at the same time, if needed.</p>
Nonproduction environments	<pre>java -jar connectivityagent.jar</pre> <p>You can restart the agent as a background process. See Run the Connectivity Agent Installer as a Background Process on Linux Systems.</p>

Create a Connection with an Agent Group

After you install the on-premises connectivity agent, you can create a connection that uses the agent group and its associated on-premises agent. Only agent groups whose monitoring status is green on the Monitor agents page and which are not associated with an adapter can be selected.

1. In the navigation pane, click **Design**, then **Connections**.
2. Click **Create**.

See [Create a Connection](#).

3. Select the adapter to configure as an invoke (target) endpoint. See [About Creating Hybrid Integrations](#).
4. Configure the connection properties and security properties.
5. In the **Access type** section, select **Connectivity agent**, then click **Associate agent group** to select the agent group to associate with the adapter. This enables you to access your on-premises applications.
6. Select the agent group to use with this adapter, and click **Use**.
7. Click **Test**. This test processes the ping command on the on-premises instance when the connection is associated with a connectivity agent.
8. Click **Save**.
9. Create an integration in which you use the adapter.
10. Activate the integration. See [Activate an Integration](#).
11. Invoke the integration.

 **Note:**

If you receive the following error, a connection timeout has occurred. The request may be slow. In that case, process the request again. You can also view the agent logs to see what may be causing the request to not process.

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.

Upgrade the Connectivity Agent Automatically

When a new version of the on-premises connectivity agent becomes available, your host is automatically upgraded with the latest version. When Oracle Integration is upgraded, the agent is upgraded within a four-hour window. There is no separate alert for agent upgrade. There is no downtime or interruption of service for in-process integrations that use the agent. If there is any failure for in-process integrations using the agent, those integrations may require resubmission.

- [Connectivity Agent Upgrade Process](#)
- [View Connectivity Agent Restart Progress on Windows](#)

Connectivity Agent Upgrade Process

Connectivity agent upgrade occurs as follows:

1. A check is made of the version of the agent installed on your on-premises host.
2. If the agent version on your host is older than the latest available version, the new version is downloaded to your host.
3. The downloaded ZIP file is extracted.
4. A backup directory is created.
5. A backup copy is made of your existing installation in the new backup directory.

6. Older artifacts are replaced in the `agenthome` directory of your installation.
7. The endpoints are quiesced.
8. The agent is shut down and restarted.
You are notified of upgrade success.

 **Note:**

After agent upgrade occurs, the agent fails to initialize the SAP JCO libraries because these are native libraries and can only be loaded once per JVM. In the case of automatic agent upgrade, the agent restart reloads the classes (after agent upgrade) without shutting down the JVM. As a workaround, a manual restart of the agent is required. See [Restart the Connectivity Agent](#).

View Connectivity Agent Restart Progress on Windows

The connectivity agent is automatically restarted after upgrade completes. During the restart on Windows, the connectivity agent runs in the background.

To view connectivity agent restart progress, you can look in the PID file (`oic_connectivity_agent/pid`) for the process ID, run the `jps` command from the prompt (as shown below), or view the connectivity agent upgrade logs located in `agenthome/logs/agent-diagnostic0.log` and `agenthome/logs/agent-upgrade.log`.

```
C:\OIC\Agent\Agentwindows\ADT\oic_connectivity_agent>jps
12504 OracleIdeLauncher
13944
17960 connectivityagent.jar
19496 Jps
4104 OracleIdeLauncher
```

Recover the Connectivity Agent in Case of Host Failure

If the host on which the connectivity agent is running fails and makes your environment unavailable, follow these steps to recover the connectivity agent. The failed host can be an on-premises machine or in Oracle Cloud Infrastructure Compute.

 **Note:**

- Perform regular backups of `AGENT_INSTALL_DIR`.
- Ensure that the regular backups are not on the same physical host on which the agent is currently running.
- Ensure that backups are available in the event of an agent host failure.

If your host fails and makes the environment unavailable, do the following;

1. Copy the `AGENT_INSTALL_DIR` directory from your last backup to the new host.
2. Change directories to the `AGENT_INSTALL_DIR` directory.

3. Restart the connectivity agent.

```
java -jar connectivityagent.jar
```

 **Note:**

If you use third party JAR files, ensure that the same JARs are available under `AGENT_INSTALL_DIR/agenthome/thirdparty/lib`. If not, add them to the lib directory.

 **Note:**

- To ensure the integrity of the backups, always create a new backup if integration activations/deactivations have occurred since the last backup.
- The agent instance name is from the backed up configuration and is mapped to the current host. If backup and recovery are performed on a different host, the agent instance name does not change to reflect the new host on which the agent is running.

Use the Connectivity Agent in High Availability Environments

You can use the connectivity agent in high availability environments with Oracle Integration. You install the connectivity agent twice on different hosts. The connectivity agents can scale horizontally, thereby providing all the benefits of running multiple agents for an agent group. This results in increased performance and extends failover benefits.

- [Prerequisites](#)
- [Install the Connectivity Agents](#)
- [Features and Considerations](#)

Prerequisites

Ensure that both agent instances can access the same endpoints. For example, agent 1 on host 1 and agent 2 on host 2 must both be able to access the same endpoint (for example, a Siebel system).

Install the Connectivity Agents

1. Create a new agent group. You cannot use an existing group.
2. Download and install the first agent on one host. Ensure that you specify the agent group identifier of the new agent group.
See [Download and Install the Agent](#).
3. Follow the same steps to download and install the second agent on a second host. That way, if one host goes down, agent processing continues running on the other host.
 - Use the same JAR file that you downloaded for the first installation.
 - Use the same `InstallerProfile.cfg` file settings. You can also copy the file from one host to the other.

 **Note:**

Specify the same agent group identifier and Oracle Integration URL in the `InstallerProfile.cfg` file as with the first installation.

4. After installation completes, go to the Agents page and note that two agents are associated with the same agent group.
5. Go to the Monitor agents page and note the same information. In this case, both agent instances are running.

Features and Considerations

- You can install multiple agents on the same host. However, to use high availability capabilities, install the second agent on a second host.
 - Create a horizontal cluster to achieve high availability. Installing the agent on the same virtual machine (VM) does not guarantee high availability.
- You cannot have more than two agent instances per agent group. Attempting to include a third agent instance in the same group during installation results in an error.
- While you can install up to two agent instances per agent group, be aware of the current restriction with agent high availability when working in tandem with automatically upgraded agents. To ensure that both agent instances do not upgrade at the exact same time, it is recommended that you start each instance with a small time gap. The time gap can be as little as 10 minutes. This ensures that there is no outage due to both agent instances being upgraded at the same time.

Delete an Agent Group

You can delete an agent group that is not currently associated with a running connectivity agent instance. Deleting the agent group deletes the internal artifacts that the agent relies on after agent group creation. The OAuth client application in Oracle Identity Cloud Service is also deleted.

1. Before deleting an agent group, ensure that you terminate the running agent at the operating system command prompt:

```
kill -9 agent_PID_number
```

The `agent_PID_number` can be obtained from the `AGENT_INSTALL_DIRECTORY/pid` file.

2. In the navigation pane, click **Design**, then **Agents**.
3. Find the agent group to delete. If the number count for the agent group is zero, you can delete the agent. Otherwise, you must click the number (it can be 1 or 2) and first delete those agent instances. If any connections are using the agent, you cannot delete the agent instance.
4. Hover over the agent to delete.
5. Click **Actions . . .**, then select **Delete**.
6. Select **Delete** when prompted to confirm.

Agent Behavior in a Decommissioned Instance or HTTP 404/401 Error Response Codes

The connectivity agent continuously seeks work to process. If the Oracle Integration instance is decommissioned or if all requests to the instance from the agent fail continuously with an HTTP error code of 401/404 for 24 hours, the agent terminates the poller threads. This halts all message processing. When the conditions leading to the error have been resolved, the agent must be restarted manually.

See [Restart the Connectivity Agent](#).

10

Use JavaScript Libraries in Integrations

You can add JavaScript libraries into integrations.

Topics:

- [Use Libraries to Manage Integration Functions](#)
- [View Library Details](#)
- [Edit Library Function Values](#)
- [Determine Which Integrations Use a Library Function](#)
- [Delete a Library](#)
- [Import a Library File](#)
- [Export a Library File](#)
- [Update JavaScript Files](#)
- [Use Library Functions in Integrations](#)

Use Libraries to Manage Integration Functions

You can include JavaScript libraries in integrations. JavaScript libraries provide a means to create and organize JavaScript for use in integrations. Library functions are available for you to drag into your integrations.

You can use JavaScript library functionality to:

- Create a new JavaScript library
- See a list of all created JavaScript libraries
- View the details of a JavaScript library
- Edit the details of a JavaScript library
- Determine which integrations are using library functions
- Delete a JavaScript library
- Import functions
- Download a library source file to your desktop
- Add library functions in an integration

JavaScript libraries are not provided out of the box. You create your own libraries. However, the Expression Builder and the mapper automatically provide built-in functions. See Add Functions, Operators, and XSLT Statements in *Using the Oracle Mapper with Oracle Integration 3*.

View Library Details

You can view details about a library without changing any information.

1. In the navigation pane, click **Design**, then **Libraries**.
2. Hover your cursor over the library to view, then click **View** .

The files and their constituent functions are displayed. You can expand and contract the list of functions and search and filter the list. To view information about a function, click the function. To view the source code, click the file name and download the code to your desktop. You can also click the file name to annotate the entry with your comments.

Edit Library Function Values

You can edit JavaScript library functions and files in Oracle Integration. The JavaScript library can consist of a single file or multiple JavaScript files in a JAR file.

1. In the navigation pane, click **Design**, then **Libraries**.

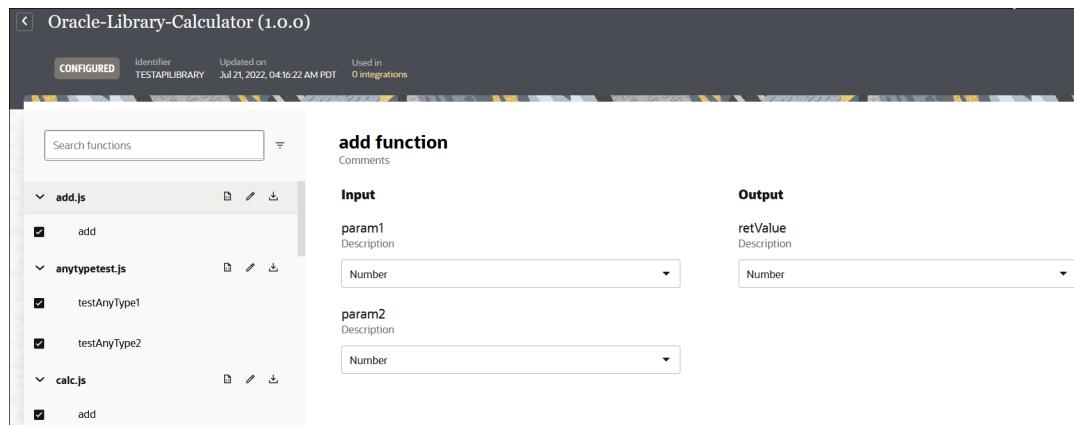
2. Either click the library name or click **Edit** .

The files contained in the library and their constituent functions are displayed. You can expand and contract the list of functions, and you can search and filter the list.

3. To edit information about the input and output values of a function, click the function.

If the function is in use in an integration, you are warned about it and the function is presented in view-only mode.

If the function is available for editing, you can set the data types of the function's input and output values, defining them as **Boolean**, **Number**, or **String**.



4. When done, click **Save**.

Determine Which Integrations Use a Library Function

You can determine which integrations are using functions in a given library.

1. In the navigation pane, click **Design**, then **Libraries**.
2. Go to the row of the library to check.
3. In the **Usage** column, click the number. This value indicates the number of integrations that use functions in this library.

When you click the number, the Library is in use dialog is displayed. It lists the names of the integrations that use the library's functions. This dialog is also invoked if you attempt to edit or delete a library that is currently in use.

 **Note:**

If you attempt to activate an integration that includes a function that is not completely configured, an error message is displayed in the banner. You must complete configuration of this function before you can activate the integration. See [Activate an Integration](#).

Delete a Library

You can delete a library not currently in use by an integration.

1. In the navigation pane, click **Design**, then **Libraries**.
2. Hover over the library to delete.
3. Click **Actions ...**, then select **Delete**.

If any of the library's functions are used by an integration, the Library is in use dialog is displayed and you are prevented from deleting the library. You must remove the functions from the integration before you can delete the library.

Import a Library File

You can import a `.zip`, `.jar`, or `.js` library file into Oracle Integration. The ZIP file must contain only the library JAR and metadata XML file.

1. In the navigation pane, click **Design**, then **Libraries**.
2. Click **Import**.
The Import panel opens.
3. Click **Drag and Drop** to select a file or drag one to this field.
4. Make updates, when necessary, then click **Save**.

Write JavaScript

JavaScript provides an alternative to using the mapper and assign action to enrich data, but not to develop application logic.

Refer to your JavaScript document when writing or copying JavaScript code to run in a development (sandbox) environment. See [JavaScript](#). This documentation provides basic syntax and available constructs. Because the script runs in a development environment, network, file, and Java APIs are not available.

Code Format

Name function return parameters as shown in the following example. You must configure the return parameter type in the metadata user interface only if the `return` parameters are named. Consider the following code:

```
function add ( param1, param2 )
{
    return param1 + param2;
}
```

Although the above syntax is valid, the `return` parameter does not appear in the metadata user interface for configuration. For that to occur, the code must return a named parameter. Create a new variable and assign return values before the variable is returned. For example:

```
function add ( param1, param2 ) {
    var retValue = param1 + param2;
    return retValue;
}
```

Function Within Another Function

Functions defined within another function do not appear in the metadata user interface for configuration. You may need to refactor the code if you need to access the function from the callout or mapper.

In the following code, `foo` is defined within `parseDate` and does not appear in the metadata user interface for configuration.

```
function parseDate(d) {
    function foo(d) {
        if(typeof d === 'string') { return Date.parse(d.replace(/-/g,'/')); }
        if(!(d instanceof Array)) { throw new Error("parseDate: parameter must
be arrays of strings"); }
        var ret = [],k;
        for(k=0;k<d.length;k++) { ret[k] = foo(d[k]); }
        return ret;
    }
    var retVal = foo(d);
    return retVal;
}
```

Extensions

Oracle Integration provides a set of extension functions organized under the following namespace:

Function	Example
uuid functions:	<pre>function uidv1() { var uidv1_result = oic.uuid.uidv1(); return uidv1_result; } function uidv4() { var uidv4_result = oic.uuid.uidv4(); return uidv4_result; }</pre>
checksum functions:	<pre>function checksum_sha(inputStr) { var sha_result = oic.checksum.sha(inputStr,"sha"); return sha_result; } function checksum_sha256(inputStr) { var sha256_result = oic.checksum.sha256(inputStr,"sha-256"); return sha256_result; } function checksum_sha512(inputStr) { var sha512_result = oic.checksum.sha512(inputStr,"sha-512"); return sha512_result; } function checksum_md5(inputStr) { var md5_result = oic.checksum.md5(inputStr,"md5"); return md5_result; }</pre>
base64 functions:	<pre>function base64_encode(input_encode_str) { var encode_result = oic.base64.encode(input_encode_str); return encode_result; } function base64_decode(input_decode_str) { var encode_result = oic.base64.decode(input_decode_str); return encode_result; }</pre>

Function	Example
crypto functions: oic.crypto hmacsha256	<pre>function generateHashCode(hmacString_value, hmacKey_value) { var hashCode_value = oic.crypto.hmacsha256 (hmacString_value, hmacKey_value); return hashCode_value; }</pre>

Export a Library File

You can export a library to a source file (.jar or .js) on your desktop.

1. In the navigation pane, click **Design**, then **Libraries**.
2. Hover over the library to export.
3. Click **Actions ...**, then select **Export**.

Update JavaScript Files

You can update a JavaScript file directly in Oracle Integration or in a different editor. Updating the JavaScript file directly in Oracle Integration eliminates the need to export the file, edit it in a different editor, and then re-import it into Oracle Integration. The JavaScript library can consist of a single file or multiple JavaScript files in a JAR file.

Note the following guidelines for updating a JavaScript file:

- Functions (within a library) used in integrations must be available in the updated JavaScript file.
- The signatures (parameters and function name) for these functions must remain intact to prevent runtime issues.
- The addition of new functions is allowed.
- Active integrations using new or updated functions must be reactivated.
- You can remove functions used in integrations.

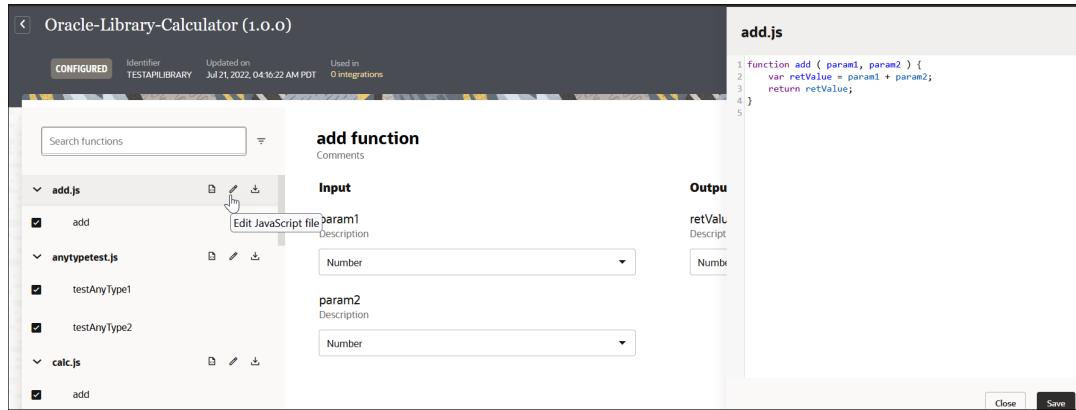
The following changes make an updated JavaScript file invalid:

- Changing the name of a function used in an integration.
- Changing the parameter definitions of functions used in an integration.
- Structural issues with the JavaScript file.
- The number of functions exceeds 500.

Update a JavaScript File Directly in Oracle Integration

You can also edit the JavaScript file code. Options are also provided for viewing the file contents in read-only mode or downloading the JavaScript file.

1. If you want to edit the JavaScript file code, click **Edit** . An editor slides open on the right to display the content of the selected JavaScript file.



- Add, remove, and update content, when necessary. For example, add a second function (for this example, named `add_xy`).

```
function add ( param1, param2 ) {
    var retVal = param1 + param2;
    return retVal;
}

function add_xy ( x, y ) {
    var retVal = x + y;
    return retVal;
}
```

- Click **Save** when complete. This action adds a second function called `add_xy` in the left pane. The editor performs basic validation, but is not a complete JavaScript editor.

Update a JavaScript File in a Different Editor

You can still update a JavaScript file in a different editor.

- In the navigation pane, click **Design**, then **Libraries**.
- Hover over the library to update.
- Click **Actions ...**, then select **Update**.

For example, assume the library to update includes four functions, one of which (`multiply`) is currently used in an integration. The content of the library is as follows:

```
function add ( param1, param2 ) {
    var retVal = param1 + param2;
    return retVal;
}
function subtract ( param1, param2 ) {
    var retVal = param1 - param2;
    return retVal;
}
function multiply ( param1, param2 ) {
    var retVal = param1 * param2;
    return retVal;
}
function divide ( param1, param2 ) {
    var retVal = param1 / param2;
```

```
        return retValue;
    }
```

4. Click **Browse** to select the JavaScript file that includes the function updates, then click **Update**.

For this example, the JavaScript file consists of the following updates:

- Two new functions: `add_three` and `multiply_three`
- One removed function not used by any integrations: `subtract`
- A code change to the `multiply` function that is currently used in an integration

```
function add ( param1, param2 ) {
    var retValue = param1 + param2;
    return retValue;
}
function multiply ( param1, param2 ) {
    var retValue = param1 * param2 * 2;
    return retValue;
}
function divide ( param1, param2 ) {
    var retValue = param1 / param2;
    return retValue;
}
function add_three ( param1, param2, param3 ) {
    var retValue = param1 + param2 + param3;
    return retValue;
}
function multiply_three ( param1, param2, param3 ) {
    var retValue = param1 * param2 * param3;
    return retValue;
}
```

All these updates are valid and the library is successfully updated. The library edit page is displayed with the updates. Any active integrations using the `multiply` function must be reactivated.

An example of an invalid JavaScript file that *cannot* be successfully uploaded is if the `multiply` function included a parameter definition that changed (for example, `param2` to `param3`).

```
function add ( param1, param2 ) {
    var retValue = param1 + param2;
    return retValue;
}
function multiply ( param1, param3 ) {
    var retValue = param1 * param2 * 2;
    return retValue;
}
function divide ( param1, param3 ) {
    var retValue = param1 / param2;
    return retValue;
}
```

Use Library Functions in Integrations

You can add library functions to integrations.

After you create and edit your library functions, they are available from the palette or the canvas inside an integration. See [Add JavaScript Functions to an Integration with a JavaScript Action](#).

11

Manage Integrations

When integration design is complete, you can perform management tasks such as integration activation, integration testing, and other responsibilities.

Topics:

- [Activate and Deactivate Integrations](#)
- [Test Integrations in Oracle Integration](#)
- [Edit and Replace Dependent Resources in an Integration](#)
- [Invoke a Child Integration from a Parent Integration](#)
- [Manage Packages](#)
- [Import and Export Integrations](#)
- [Clone the Design-Time Metadata of an Entire Service Instance](#)
- [Refresh Endpoints for Integrations](#)
- [Set the Time Limit for Inactive Sessions](#)
- [Modify an Integration](#)
- [View the Actions and Connections in an Integration](#)
- [Clone an Integration](#)
- [Create a Draft of an Integration](#)
- [Delete an Integration](#)
- [Recover Unsaved Integration Changes](#)
- [Override Design-Time Properties in an Integration](#)
- [Use Metadata in Integrations](#)
- [Cancel a Running Integration Instance](#)
- [View the Contents of an Incomplete Integration](#)
- [Edit an Integration with Incomplete Connections](#)
- [Change the Time Zone](#)

Activate and Deactivate Integrations

After integration design is complete, you can activate and set the tracing level for your integration. You can also deactivate integrations.

Topics:

- [Activate an Integration](#)
- [Reactivate Integrations after a Connection Update](#)
- [Deactivate an Integration](#)

- Manage Tracing Levels on Integrations

Activate an Integration

Once you have completed design of your integration and resolved any errors, you can activate it to the runtime environment.

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.

Understand Restrictions About the Number of Active Integrations

The number of active integrations per Oracle Integration instance cannot exceed 800. Active integrations are defined as currently active integrations and integrations whose activations are in progress.

- If you reach 90% of this limit, the following warning is displayed:

You've *number* integrations which are either active or whose activation is in progress.

It is more than 90% of the allowed limit.

- If you reach the limit of 800, the **Activate** or **Activate & Schedule** button to activate an integration is disabled and the following warning is displayed.

You've reached the limit as there are 800 integrations which are either active

or whose activation is in progress. Deactivate or abort the activation of an integration

and try again.

If you are nearing your limit, review and delete any older integrations that are no longer required.

Activate an Integration

1. In the navigation pane, click **Design**, then **Integrations**.
 2. Hover over the integration to activate, then click **Activate** .
- The Activate integration panel is displayed.
3. Select the level of tracing appropriate to your integration. This level describes the amount of data to capture in the activity stream. The tracing level has no impact on billing.

WARNING:

The **Audit** and **Debug (Not recommended)** tracing levels are not recommended for a production environment. These options may slow down your system and also present a security risk because sensitive information from the payload is logged.

Element	Description
Tracing	<p>Select the level of tracing information to log to the activity stream.</p> <ul style="list-style-type: none"> • Production: Logs all actions with the exception of loops and invoke/logger actions inside of loops (up to 1000 iterations) to the activity stream. Data is retained in the activity stream for 32 days. <p> Note:</p> <p>Oracle Integration for Healthcare data is retained for 184 days.</p> <ul style="list-style-type: none"> • Audit: Logs the same data to the activity stream as the Production option, along with wire payloads of triggers and invokes. Data is retained in the activity stream for eight days. Wire payloads are the payloads sent and received over the wire by Oracle Integration while communicating with third-party applications. These communications either happen through triggers (where Oracle Integration receives the initial triggering message) or invokes (where Oracle Integration sends and receives data). The data in these payloads is untranslated and in the same format understood by the external applications/services that send and receive these payloads. • Debug (Not recommended): Logs the same data to the activity stream as the Production and Audit options along with all payloads and all actions inside loops (up to a 1000 iterations). Data is retained in the activity stream for 24 hours. After that time period elapses, all payload, activity stream, and business identifier data is purged and a minimal amount of instance information is retained for 32 days. Note the following: <ul style="list-style-type: none"> – After 24 hours, Debug (Not recommended) is automatically reset to Production. This reset cannot be bypassed. – Explicitly reset the tracing level to Production or Audit when debugging is complete. Do not rely on the 24-hour automatic reset. The actual reset may take longer than 24 hours if no messages are processed by a given integration. <p>See View Minimal Details About Debug Tracing Level Instances for 32 Days.</p> <p>See Activity Stream Details.</p>
Allow to run again	This check box only appears for REST Adapter-triggered and AS2 Adapter-triggered integrations. Select this check box if you want to replay an instance of this integration during runtime. See Replay Integration Instances .
Enable payload validation	Select to validate inbound JSON payload syntax during activation of REST Adapter trigger connection-based integrations. Validation consists of checking for duplicate keys in the payload. If you select this option, it remains selected by default. For example, if you later deactivate and reactivate the integration, the check box remains selected. You must explicitly deselect this check box if you no longer require its use.
Edit/Add Schedule	Select to add or edit a schedule for this integration now. This invokes the page for adding or editing an integration schedule.

4. If integration activation is unsuccessful, the status of the integration changes to **Failed** in the **Status** column. Click the **Open Details** icon to see the error.

If your integration includes a function that is not completely configured, an error message is displayed in the banner. You must complete configuration of this function before you can activate the integration. Click inside the integration and note the following errors/warnings:

- An error icon is displayed on the function call action that uses the incomplete function. The **Error** panel on the right side of the integration canvas provides specific details about the incomplete function.
- A warning icon is displayed on the mapper that uses the inputs and outputs of this function. After completing function configuration, you must verify the input and output mappings before activating the integration.

If activation is successful, the status of the integration changes to **Active** in the row.

5. If integration activation is successful, an **Active** message is displayed in the **Status** column.

Additional messages can also be displayed:

- If you selected **Audit** during activation, an **Audit Tracing** message is displayed after **Active**.
- If you selected **Debug (Not recommended)** during activation, a **Debug Tracing** message is displayed after **Active**.

6. For REST Adapter trigger-based integrations, click **Actions ...**, then **Run** to display the Configure and run page to run, test, and track instances for this integration. See [Test Integrations from Outside the Integration Canvas](#).

Reactivate Integrations after a Connection Update

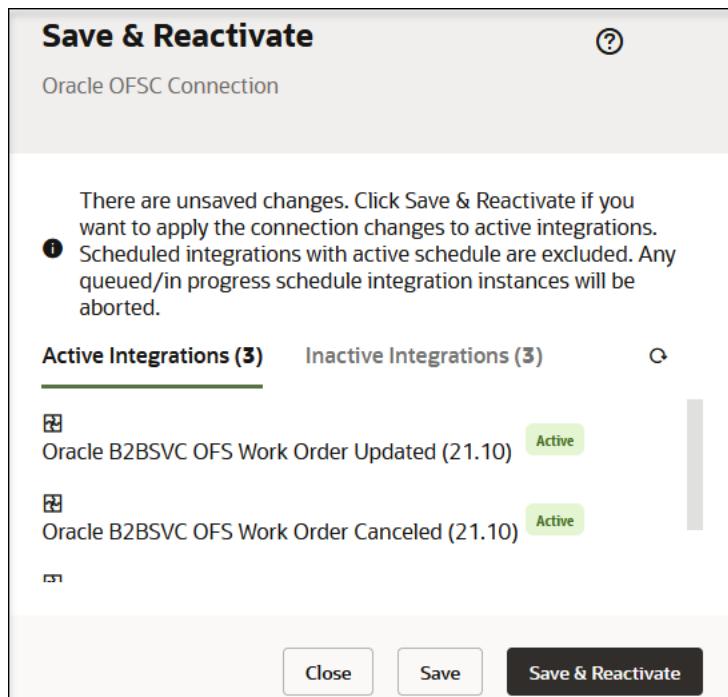
You can reactivate all active integrations together that use the same adapter connection after a connection update. This eliminates the need to manually deactivate and reactivate each integration individually to reflect the updated adapter connection. This functionality works for all adapters.

 **Note:**

A maximum of 50 integrations can be reactivated after a connection update.

1. In the navigation pane, click **Design**, then **Connections**.
2. Find and update the necessary connection. The number of integrations that use that connection is displayed.
3. Click **Test**, then click **Save**.

The list of active and inactive integrations that use the updated connection is displayed on the right.



4. Click **Save & Reactivate** to apply the connection changes to all active integrations. Note the following:
 - You cannot select individual integrations to reactivate; all are reactivated.
 - Any active schedule integrations are temporarily paused and enter a failed state. The schedule restarts after reactivation completes. However, schedule integrations that are currently in-flight enter a failed state and are discarded.
 - Any integrations using this connection that are currently listed as inactive remain in that state.
 - If you click **Save**, your connection changes are saved, but no integrations are reactivated.
 - Deactivation of an integration that subscribes to business events does not unsubscribe the business events.

The page is refreshed to show integration activation progress (for example, the number of integrations that have been reactivated and the number of integrations remaining to be reactivated).

5. Click **Refresh**  periodically to view progress.
When complete, details about the reactivated integrations are displayed.
6. Click **Cancel** to close the page.
7. If any integration reactivations failed, go to the Integrations page to investigate.

Deactivate an Integration

You can deactivate an integration in Oracle Integration to stop it from processing any new messages. If you want to modify an active integration, you need to deactivate it first. If the

integration contains a business event subscription, a message is displayed asking if you want to delete the event subscription.

Deactivation is equivalent to undeployment of a project, which means all existing history, monitoring, and runtime data are lost. Oracle Integration 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 navigation pane, click **Design**, then **Integrations**.
2. Go to the row of the integration you want to deactivate.

To view only active integrations, select **Active** in the **Status** field of the **Filter**  filter. You can also filter by integration name, time last updated, integration creator, status (all, draft, configured, active, activating, and failed), type (all, developed, accelerator type, and recipe), or style (all, application, schedule, and event) to narrow down the list.

3. Click **Deactivate**  to deactivate the integration.
4. Click **Deactivate** in the dialog that appears.

 **Note:**

Deactivation of an active, schedule integration is not allowed. You must stop the schedule, then retry.

Integration deactivation can take time to complete after you initiate the request. During this time period, there may still be in-flight or queued requests being processed. These processing instances fail when the corresponding integration artifacts and resources are removed and undeployed.

You can observe different processing errors for the failed instances depending on the point of processing during the deactivation. You can also observe the failed instances occurring after the deactivation request.

If deactivation fails and the integration state is displayed as **Deactivation Failed**, retry the deactivation. An integration in this state cannot be edited or activated.

When an integration is successfully deactivated, all its data get undeployed from runtime. This results in nonterminal instances (which are visible on the Instances and Errors pages) getting aborted. Instances in queued, in progress, in error (recoverable type), and other states all move to the aborted state. The existing terminal instances (succeeded, aborted, and in error (nonrecoverable type) remain unaffected.

Deactivate an Integration with Business Events

If the integration to deactivate contains a business event subscription, a message is displayed asking if you want to delete the event subscription. If you select to delete the event subscription, the integration does not receive any events after it is reactivated.

Deactivate Integration



ShopifyCreateOrderEvent (1.0)

Delete Event Subscription: It will delete the subscription associated with the integration.

Delete Event Subscription

Cancel

Deactivate

If you do not want to delete the event subscription, the events in this integration are resent if the integration is activated within six hours.

Manage Tracing Levels on Integrations

You can change the tracing level on an active integration without having to deactivate and reactivate it. You can also set the tracing level globally for all integrations.

⚠ Caution:

Changing tracing to the audit or debug level impacts performance. Oracle strongly suggests that you don't enable the debug or audit option in production environments. Including the payload in tracing impacts the performance of your system and may violate your company's data retention rules by storing sensitive information in the payload trace. Understand the impact of the various options before making a selection. See [Activate an Integration](#).

- [Change the Tracing Level on an Active Integration](#)
- [Set the Tracing Level Globally for All Integrations](#)

Change the Tracing Level on an Active Integration

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the active integration on which to change the tracing level.
3. Click **Actions ...**, then select **Configure activation**.
4. Configure the tracing level.
5. Click **Save**.

The tracing level is changed in the active integration.

The message under **Active** in the **Status** column changes for this integration. For example, if you changed the tracing level from **Production** to **Audit**, the message **Audit Tracing** is displayed after **Active** once integration reactivation completes.



Set the Tracing Level Globally for All Integrations

1. In the navigation pane, click **Settings**, then **Tracing**.
2. Select an option to configure the tracing level, and click **Save**. The tracing level has no impact on billing.

Element	Description
Integration Level	Select to enable or disable tracing at the individual integration level on the Integrations page. This enables you to select Actions ... and then Tracing for an individual integration on the Integrations page and select to disable tracing or enable tracing and include the payload in the integration. This is the default selection.
Global tracing on at audit level	Select to globally enable tracing at the audit level for all activated integrations. After selecting and saving this option, you cannot enable or disable tracing at the individual integration level on the Integrations page. Two other changes occur: <ul style="list-style-type: none">• When you access the Integrations page, a message is displayed indicating that tracing is enabled at the audit level for all active integrations and you cannot enable or disable it at the individual integration level.• If you select Actions ..., there is no Tracing option available for selection. To enable or disable tracing at the individual integration level again, you must select Integration level , then Save .
Global tracing on at debug level	Select to globally enable tracing at the debug level for all activated integrations. After selecting and saving this option, you cannot enable or disable tracing at the individual integration level on the Integrations page. Two other changes occur: <ul style="list-style-type: none">• When you access the Integrations page, a message is displayed indicating that tracing is enabled at the debug level for all active integrations and you cannot enable or disable it at the individual integration level.• If you select Actions ..., there is no Tracing option available for selection. Note: Regardless whether debug tracing is set globally or on a single integration, the tracing level is automatically reset to production after 24 hours. To enable or disable tracing at the individual integration level, you must select Integration level , then Save .

Element	Description
Global tracing off	<p>Select to globally disable tracing for all integrations. After selecting and saving this option, you cannot enable tracing at the individual integration level on the Integrations page. Two other changes occur:</p> <ul style="list-style-type: none"> When you access the Integrations page, a message is displayed indicating that tracing is disabled for all active integrations and you cannot enable it at the individual integration level. If you select Actions ..., there is no Tracing option available for selection.

Test Integrations in Oracle Integration

You can test REST Adapter trigger connection-based integrations and scheduled integrations in Oracle Integration. You can test the integration either from within the integration canvas or from the **Actions** menu of the integration outside of the integration canvas. This eliminates the need to use third-party software to run this type of integration.

Topics:

- [Test Integrations from Inside the Integration Canvas](#)
- [Test Integrations from Outside the Integration Canvas](#)

Test Integrations from Inside the Integration Canvas

You can test REST Adapter trigger connection-based integrations and scheduled integrations from inside the integration canvas in Oracle Integration.

This capability lets you automatically activate your integration with the tracing level set to debug and specify request details from the Run integration panel inside the integration canvas.

This capability simplifies integration testing and eliminates the need to exit the canvas, manually activate the integration, set the tracing level, select **Run** from the **Actions ...** menu, and specify the request details as separate steps.

1. Design your integration.

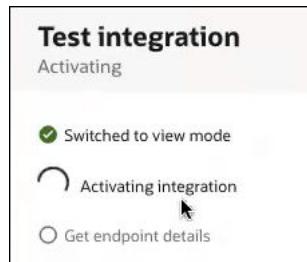
When complete, perform the following steps to activate and run your integration.

2. Click **Run test ▶** in the integration canvas bar.



The Test integration panel opens and the following actions are automatically performed:

- The integration switches to view mode.
- The integration is activated with the tracing level set to debug.
- The endpoint details are obtained.



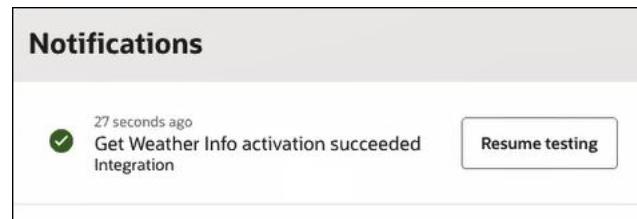
While activation tasks are in progress, you can perform other actions:

- Perform actions inside the canvas. For example, you can click **Properties**  to view integration properties. When you perform an action inside the canvas, the Test integration panel is closed. To bring back the Test integration panel and continue testing, click  in the toolbar.
- Exit the canvas to perform actions. Once activation completes, a message appears in the upper right part of the page.

Integration_name activation succeeded
Resume testing

You can click **Resume testing** to return to the canvas where you are prompted to specify request values to run the activated integration.

You can also click **Notifications**  in the Oracle Integration banner, and then click **Resume testing** to return to the canvas where you are also prompted to specify request values to run the activated integration.



If activation fails, the **Try again** option is enabled in the Test integration panel.

When these tasks complete, the panel is refreshed to show the following:

- a. For a REST Adapter trigger connection-based integration, you are prompted to specify request parameter values:

Test integration
Configure request

```
GET
/ic/api/integration/v2/flows/rest/project/TEST_FROM_CANVAS/GET_WEATHER_INFO/1.0/weatherInfo?unit=&cityName=
```

Request

URI parameters	Headers	Body
<input checked="" type="checkbox"/> unit		
<input checked="" type="checkbox"/> cityName		

- b. For scheduled integrations with defined schedule parameters, you are prompted to specify values:

Test integration
Configure schedule parameters

Instanceld

Description

Default value
100

Current value
106

City

Description

Default value
Athens

Current value
Delhi

- c. For scheduled integrations without defined schedule parameters, no additional input is required. The integration runs automatically after activation and the results/activity stream are displayed.
3. Specify the request or scheduled parameter details, and click **Run**.

The response details (for REST Adapter trigger connection-based integrations) and instance ID are first displayed, followed by the activity stream. The integration canvas switches from edit mode to runtime tracking mode.

The screenshot shows the Oracle Integration Test Integration panel. On the left, under 'Response', it displays a status of '200 OK' and the JSON body of the message: { "temp" : "20.66", "city" : "New York", "country" : "US" }. Below the body is the instance ID 'zXmu-rVXEe6ENJnweFuiAg'. On the right, the 'Activity stream' section shows a timeline of events with tracing level set to 'Debug'. The events are:

- Wed Jan 17 10:15:20.914 PM IST 2024, 390ms, Triggered by IIserviceadmin Wire Message received by Trigger GetWeatherInfo
- 10:15:21.304 PM, 17ms Message started processing
- 10:15:21.321 PM, 1ms Logger Logger1
- 10:15:21.322 PM, 3ms Message entered Switch
- 10:15:21.325 PM, 1ms Message routed through Switch Route with condition unit = 'F'
- > 10:15:21.326 PM Message received by Assignment Assignments1
- 10:15:21.330 PM, 1ms Message exited Switch
- 10:15:21.331 PM, 9ms Message received by Data Mapper

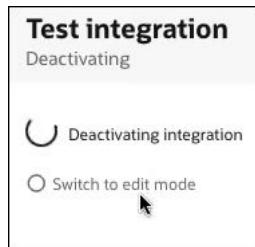
At the bottom of the activity stream are 'End test' and 'Run again' buttons.

4. Perform runtime monitoring details, such as expanding sections of the activity stream, viewing the business identifiers that were passed, tracing the message delivery through the integration, and so on.
5. If you want to run the integration again, click **Run again**.
6. Specify the request or scheduled parameter details again, and click **Run**.
7. When testing is complete, click **End test** at the bottom of the activity stream or **End test** in the integration canvas bar.



The Test integration panel opens and the following actions are automatically performed:

- The integration is deactivated.
- The integration switches back to edit mode.



Test Integrations from Outside the Integration Canvas

You can test application integrations designed with a REST Adapter trigger connection and scheduled integrations from the **Actions** menu of the integration in Oracle Integration.

See [Run a Schedule Integration Immediately](#).

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the REST Adapter trigger connection-based integration to run, then click **...** and select **Run**.

The Configure and run page is displayed with the following sections:

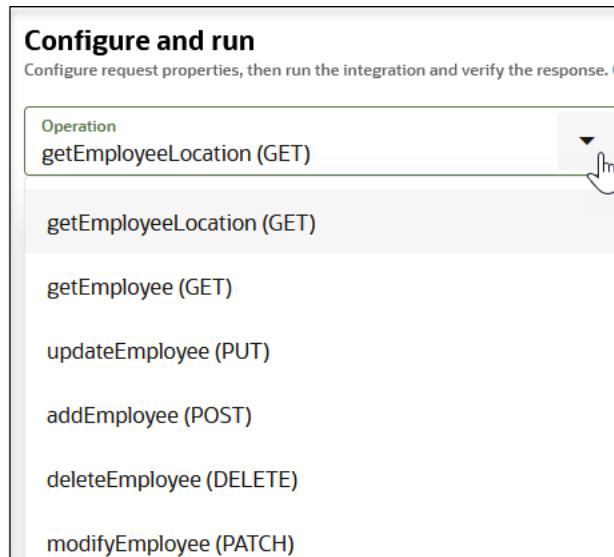
- [Operation Section](#)
- [Request and Response Sections](#)

The **Operation** and **Request** sections include the endpoint's metadata.

Operation Section

If the REST Adapter trigger connection-based integration is configured with multiple operations, the **Operation** section is displayed. This section contains the **Operation** list for selecting the appropriate operation, along with the HTTP method and relative URI (for the selected operation in case of multiple operations). If your integration contains only a single operation, this list is not displayed.

1. Select the available operation to test.



2. View the following links.

- **Endpoint Metadata:** Displays endpoint description details, such as the Swagger path, the OpenAPI path, how to run the integration, the operation name, and the endpoint URL.
- **Track Instances:** Displays the Instances page, which shows the details of runs of this instance.

3. Enter a value in the **Request** section and click **Run** to invoke the integration (for this example, a value is specified for **id**).

Integration has been invoked with instance id *id_number*. Please check the Response section for more details.

The Activity Stream panel opens and the **Response** section shows a status value of 200 OK.

The screenshot shows the Oracle Integration interface. On the left, the 'Configure and run' section is visible, showing an operation 'getEmployeeLocation (GET)' and a request URI '/kc/api/integration/v1/flows/rest/SAMPL_MULTI_VERBS_AND_RESOURCES_INVO/1.0/emp/loc?id=4566'. A checkbox labeled 'id' is checked, and its value '4566' is entered in the input field. On the right, the 'Activity Stream' panel displays a timeline of events for an instance with ID 'Qw-7DPGYEeyZT5XtxaFwxw'. The events are:

- Tue Jun 21 12:28:00 407 PM PDT 2022 Message processed by Trigger Pick Receive for Operation EmployeeAPI_getEmployeeLocation 146ms
- 12:28:00.553 PM Message started processing 10ms
- 12:28:00.571 PM Data Mapping completed 4ms
- 12:28:00.572 PM Message sent to reply 5ms
- 12:28:00.580 PM Reply to EmployeeAPI_getEmployeeLocation completed

4. Click the instance ID in the **Response** section to access the Instance Details page for this instance.

The screenshot shows the 'Instance Details' page for instance ID '4566... | kkr53PGZEyedSgv7Cg1NGg'. The page displays the integration flow diagram and the 'Activity Stream' panel. The integration flow starts with a 'Trigger EmployeeAPI' node, which branches into four parallel 'Pick' nodes: 'modifEmployee', 'updateEmployee', 'getEmployee', and 'getEmployeeLocation'. Each 'Pick' node leads to a corresponding 'Map' node ('modifEmployee', 'updateEmployee', 'getEmployee', and 'getEmployeeLocation'), which then leads to a green circular end node. The 'Activity Stream' panel on the right shows the same sequence of events as in the previous screenshot, indicating successful processing of the message.

Request and Response Sections

The **Request** and **Response** sections enable you to trigger and view response output for your REST Adapter trigger connection-based integrations. Links are provided at the top to access the Instances page and the endpoint metadata.

The **Request** section consists of the following fields:

- **URI Parameters:** Shows the list of expected path (or template) and query parameters. You can enter data in this field.
- **Headers:** Shows the custom headers, including **Accept** based on the integration configuration.
- **Body:** Enables you to view the message body contents or upload a file. Support is provided for uploading multipart payloads or for uploading nontextual payloads such as PDF or image files.
 - If the request operation does not have a body, the following message is displayed.

This request does not have a body

- If the request operation includes a body, its contents are automatically displayed. For example:

Configure and run
Configure request properties, then run the integration and verify the response. [Click here to read more.](#)

POST
/ic/api/integration/v1/flows/rest/GEN3_MRK_APPS3_INBOUN_165546/1.0/users

Request

URI Parameters	Headers	Body	cURL	Integration Properties
<input checked="" type="radio"/> Text <input type="radio"/> File <pre> 1 { 2 "event_ID": "Event ID", 3 "fn": "First Name", 4 "email": "Email Address" </pre>				

- If you want to browse for and upload a binary or nontextual payload (such as a PDF or image file), click **File**. For this example, an image file is uploaded.

Request

URI Parameters	Headers	Body	cURL	Integration Properties
<input type="radio"/> Text <input checked="" type="radio"/> File <input type="button" value="Browse..."/> oic3_adapters.png				

- For endpoints with multipart payloads, a form is displayed under the **Body** field that can be used to provide a value for each part in the multipart body. Two multipart attachments types are supported in the request payload:
 - * multipart/form-data

- * multipart/payload with structured data such as JSON and XML:
 - * multipart/form-data
 - * multipart/mixed

For the following example, two headers for multipart/form-data are visible under the **Headers** tab.

Header	Value
Accept	application/json
Content-Type	application/json

Use cases for both multipart attachment types are provided. See [Test REST Adapter Trigger-Based Integrations with Multipart Attachments on the Configure and run page](#).

- **cURL:** Enables you to copy the equivalent curl command. The curl command is generated based on the endpoint's metadata and your provided input.
- **Integration Properties:** Any defined integration properties are visible. You can configure integration properties outside of the regular integration design. At runtime, these integration properties take effect and override the values you configured at design time. See [Override Design-Time Properties in an Integration](#).

1. Click **Run** to invoke the integration.

The Activity Stream panel for this instance is displayed on the right side of the page. You can also access the instance in the Instances page by clicking **Track Instances** above the operation.

Event	Time	Duration
Wire Message received by Trigger echoRequest	Tue Jun 21 01:51:43 2022	302ms
Message started processing	01:51:43.409 PM	2ms
Logger logMessage	01:51:43.411 PM	2ms
Data Mapping completed	01:51:43.415 PM	4ms
Message sent to reply	01:51:43.417 PM	2ms
Reply to echoRequest completed	01:51:43.418 PM	

2. Check the **Response** section for response details.

The **Response** section consists of the HTTP status of the headers (200 OK for this example) and instance ID of the triggered integration that links to the Instance Details page. After this are the response body and headers included in the body.

3. Scroll to the right of **Status** and note the instance ID link. This link takes you to the Instance Details page for this instance. See [Track Integration Instances](#).

Edit and Replace Dependent Resources

You can edit dependent resources (connections, libraries, lookups, and PGP keys) in a single integration or all integrations in a package with a wizard that's accessible from the Configuration Editor page. You can also replace connections and PGP keys unless they are included in business and technical accelerators or the integration is active, locked, in process of activation, or in process of deactivation.

Only the administrator user can edit PGP keys. You can also update integration property values for integrations in which properties have been defined, edit schedules in existing schedule integrations, or add new schedules to schedule integrations.

The Configuration Editor Wizard validates your selections as follows:

- When you move to the next step in the wizard (for example, from connections to certificates), validation is performed on the previous step (connections). If validation succeeds, a green icon with a check mark is displayed on the previous step. If validation fails, a warning icon is displayed on the previous step.
- When you access the Configuration Editor page, the connections step in the wizard is shown by default. If you jump directly to the activation step from the connections step, validation runs on all the steps.
- For a simple integration without any dependencies, there is only a single activation step in which validation runs and provides the correct status for the step.
- Validation runs to check whether *all* resources under a step are in the necessary state and not just a single resource. For example, if all the connections under the connections step are in a configured state, the step status evaluates to a green icon with a check mark. Otherwise, a warning icon is displayed.

Topics:

- [Edit and Replace Dependent Resources in a Package](#)
- [Edit and Replace Dependent Resources in an Integration](#)
- [Unlock Dependent Resources](#)

Edit and Replace Dependent Resources in a Package

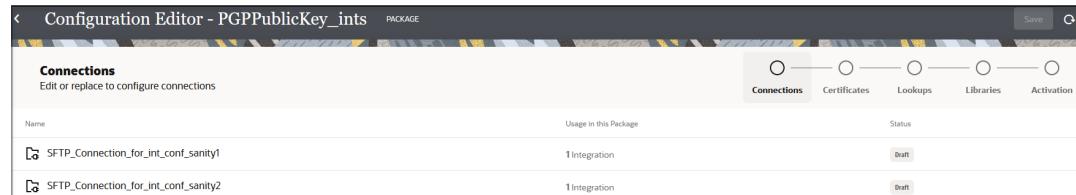
You can configure the dependent resources used by all integrations in a specific package.

Editing a dependent resource impacts all integrations that use that resource. Replacing a dependent resource only impacts the integrations in the specific package.

1. In the navigation pane, click **Design**, then **Packages**.
 2. Select the package in which to edit or replace resources in either of two ways:
 - a. In the banner, click **Import**.
 - b. Browse for and select the PAR file when prompted.
 - c. Click **Import and Configure**.
- Or:
- a. Hover over the package.

b. Click **Configure resources**

The Configuration Editor page shows the package name in the banner and a wizard that lists the steps for each resource type included in the package (connections, certificates (PGP keys), lookups, or libraries).



3. For connections, click **Edit** to edit the connection configuration and security properties or **Replace**



to replace the connection with a different one.

Note the following connection replacement guidelines:

- Only connections of the same role type (trigger, invoke, or trigger and invoke) are displayed for selection. You can only replace a connection resource with another resource of the same role.
- The connection status must be displayed as **Configured**. You cannot replace a connection with a connection whose status is **Draft**. If a compatible connection resource does not exist, a message is displayed.
- You cannot replace connections used in integrations with the following state:
 - Active
 - Activation in progress
 - Deactivation in progress
 - Locked
- If you replace a connection, a **Revert** icon is displayed in the row that enables you to change back to the previous connection.

4. Click the next resource step in the wizard (for this example, **Certificates**). If the connection is not in a configured state, an error icon is displayed. Hover over the icon to show information about the error.



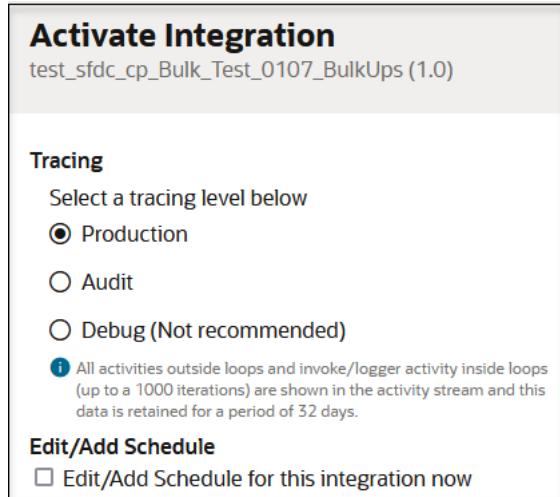
You must successfully configure the connection to activate the integration at the end of the wizard. If the connection is in a configured state, a green icon is displayed.



A PGP key must be the same type of key. For example, a public PGP key can only be replaced with another public PGP key, and not a private PGP key.

5. Click **Edit**  if you want to edit the certificate or **Replace**  to replace the certificate with a different one. If you replace a certificate, a **Revert** icon is displayed in the row that enables you to change back to the previous certificate.
6. Click the **Lookups** icon.
7. Click **Edit**  if you want to edit the lookup.
8. Click the **Libraries** icon.
9. Click **Edit**  if you want to edit the library.
If all resources are displayed as configured with a green icon and a check mark, the integration is ready for activation.
10. Click **Save**. You can also save after you update or replace a resource in each step of the wizard.
11. Hover over the row of an integration and click **Activate** . The number of active integrations per Oracle Integration instance cannot exceed 800. Active integrations are defined as currently active integrations and integrations whose activations are in progress. If you reach this limit, the **Activate** button is disabled. See [Understand Restrictions About the Number of Active Integrations](#).
You can activate business accelerators, recipes, and technical accelerators in bulk. For packages that you created, each integration must be activated individually.

The Activate Integration panel is displayed.



Activate Integration
test_sfdc_cp_Bulk_Test_0107_BulkUps (1.0)

Tracing
Select a tracing level below

Production

Audit

Debug (Not recommended)

ⓘ All activities outside loops and invoke/logger activity inside loops (up to a 1000 iterations) are shown in the activity stream and this data is retained for a period of 32 days.

Edit/Add Schedule

Edit/Add Schedule for this integration now

12. Specify the following options.
 - Select a tracing level. See [Activate an Integration](#).
 - If this is a schedule integration, you can select to edit an existing schedule or add a new schedule. See [Edit an Integration Schedule](#) or [Define the Integration Schedule](#).
 - If there are property values defined for this integration, you can select to edit them. You can configure integration properties outside of the regular integration design. At

runtime, these integration properties take effect and override the values you configured at design time. See [Override Design-Time Properties in an Integration](#).

13. Click **Activate.**

The package appears as follows:

Name	Style	Status
EncryptwithPGPPublicKey_int_conf_sanity1 (1.0)	Scheduled orchestration	Active
EncryptwithPGPPublicKey_int_conf_sanity2 (1.0)	Scheduled orchestration	Active

Edit and Replace Dependent Resources in an Integration

You can configure dependent resources used by standalone integrations that are not part of a package.

Editing a dependent resource impacts all integrations that use that resource. Replacing a dependent resource only impacts the integrations in the specific package or the standalone integration that is not part of any package.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Select the integration in which to edit or replace resources in either of two ways:
 - Hover over the integration, then click **...** and select **Configure**.
 - Select **Import**, then browse for and select the integration to import, then select **Import and Configure**.

The Configuration Editor page shows either of two views:

- If the integration you selected is part of a package, the Configuration Editor page shows the package name in the banner and a wizard that lists the steps for each resource type included in the package (connections, certificates (PGP keys), lookups, or libraries).

Name	Usage in this Package	Status
SFTP_Connection_for_int_conf_sanity1	1 Integration	Draft
SFTP_Connection_for_int_conf_sanity2	1 Integration	Draft

- If the integration you selected is not part of a package, the Configuration Editor page shows the integration name in the banner and a wizard that lists the steps for each resource type included in the integration (connections, certificates (PGP keys), lookups, and libraries).

Name	Usage	Status
SFTP_Connection_for_int_conf_sanity1	1 Integration	Draft

3. For connections, hover over the row and click **Edit**  to edit the connection configuration and security properties or



to replace the connection with a different one.

Note the following connection replacement guidelines:

- Only connections of the same role type (trigger, invoke, or trigger and invoke) are displayed for selection. A connection resource can only be replaced by another resource of the same role.
- The connection status must be displayed as **Configured**. You cannot replace a connection with a connection whose status is **Draft**. If a compatible connection resource does not exist, a message is displayed.
- You cannot replace connections used in integrations with the following state:
 - Active
 - Activation in progress
 - Deactivation in progress
 - Locked
- If you replace a connection, a **Revert** icon is displayed when you hover over the row that enables you to change back to the previous connection.

4. Click the next resource step in the wizard (for this example, **Certificates**).

If the connection selected in the previous step is not in a configured state, an error icon is displayed. Hover over the icon to show information about the error.



You must successfully configure the connection to activate the integration at the end of the wizard. If the connection is in a configured state, a green icon with a check mark is displayed.



A PGP key must be the same type of key. For example, a public PGP key can only be replaced with another public PGP key, and not a private PGP key.

5. Hover over the row and click **Edit**  if you want to edit the certificate or **Replace resource**



to replace the certificate with a different one. If you replace a certificate, a **Revert** icon is displayed when you hover over the row that enables you to revert to the previous certificate.

6. Click the **Lookups** icon.

7. Hover over the row and click **Edit** if you want to edit the lookup.
8. Click the **Libraries** icon.
9. Hover over the row and click **Edit** if you want to edit the library.
If all resources are displayed as configured with a green icon and a check mark, the integration is ready for activation.
10. Click **Save**. You can also save after you update or replace a resource in each step of the wizard.
11. Hover over the row and click **Activate** . The number of active integrations per Oracle Integration instance cannot exceed 800. Active integrations are defined as currently active integrations and integrations whose activations are in progress. If you reach this limit, the **Activate** button is disabled. See [Understand Restrictions About the Number of Active Integrations](#).

The Activate Integration panel is displayed.

The screenshot shows the 'Activate Integration' panel for the integration 'test_sfcd_cp_Bulk_Test_0107_BulkUps (1.0)'. The 'Tracing' section contains three radio button options: 'Production' (selected), 'Audit', and 'Debug (Not recommended)'. A note below states: 'All activities outside loops and invoke/logger activity inside loops (up to a 1000 iterations) are shown in the activity stream and this data is retained for a period of 32 days.' The 'Edit/Add Schedule' section has a checkbox labeled 'Edit/Add Schedule for this integration now'.

12. Specify the following options.
 - Select a tracing level. See [Activate an Integration](#).
 - If this is a schedule integration, you can select to edit an existing schedule or add a new schedule. See [Edit an Integration Schedule](#) or [Define the Integration Schedule](#).
 - If there are property values defined for this integration, you can select to edit them. You can configure integration properties outside of the regular integration design. At runtime, these integration properties take effect and override the values you configured at design time. See [Override Design-Time Properties in an Integration](#).
13. Click **Activate**.

Unlock Dependent Resources

You can directly unlock any locked dependent resources in the Configuration Editor Wizard for editing instead of navigating to the specific resource page (for example, the Connections page or Lookups page). Resources can become locked because of a system or browser crash.

If any resources are listed as locked (for this example, connections are shown), you can unlock and edit them.

Name	Usage in this Package	Status
RN_for_integration_filtering	1 Integration	Locked
RN_for_pkg_conf_long2	4 Integrations	Locked
SA_REST_ATTACH_INVOKE	1 Integration	Configured

1. Hover over the resource.
2. Click **Actions ...**, then select **Unlock** to unlock the resource for editing.
3. Click **Unlock** to confirm when prompted.

Locked integrations with unsaved changes show the words **Unsaved Changes** on the Activation step of the Configuration Editor Wizard. This example shows several integrations in the package with this status.

Name	Style	Status
integration_for_pkg_conf_long0 (1.0)	App driven orchestration	Locked
integration_for_pkg_conf_long1 (1.0)	App driven orchestration	Locked
integration_for_pkg_conf_long2 (1.0)	App driven orchestration	Locked
integration_for_pkg_conf_long3 (1.0)	App driven orchestration	Configured

1. If the locked integration does not include unsaved changes:
 - a. Hover over the integration.
 - b. Click **Actions ...**, then select **Unlock** to unlock an integration with no unsaved changes.
 - c. Click **Unlock** to confirm when prompted.
2. If the integration includes unsaved changes, two options are displayed:
 - a. Select **Unlock & Discard** to discard all unsaved changes.
or
 - b. Select **Edit & Recover** to recover all unsaved changes.
The canvas for the integration is displayed.
 - c. Select **Save** when prompted. Otherwise, click **Discard**.
You are returned to the Configuration Editor Wizard.

Invoke a Child Integration from a Parent Integration

You can invoke a co-located (child), active integration from the parent integration that you are designing through use of the local integration adapter. Co-located means the integration is

running on the same host instance or in the same domain. Upon activation and invocation of the parent integration, it invokes and consumes the co-located integration.

- [Capabilities](#)
- [Restrictions](#)
- [Best Practices for Designing Local Invoke Calls](#)
- [Invoke a Co-located Integration with a Parent Integration](#)

Capabilities

The local integration adapter is a system adapter that is not exposed on the Connections page when you create a connection. Instead, the local integration adapter is invoked by a preseeded connection. It is created when creating an integration, if the local integration connection does not exist. This adapter has no connection properties and no security policies.

The local integration adapter works as follows:

- You select an active integration from runtime (a child) to build a new integration. Note the following guidelines:
 - Active synchronous-synchronous, asynchronous with no callback, and schedule integrations are available for selection.
 - Asynchronous with callback integrations do not appear.
 - De-activated and draft integrations do not appear.
 - Schedule integrations are permitted as an invoke connection and automatically called by the **Submit Now** option.
- You can add multiple child integrations. You can also add an integration containing a child integration (embedding multiple levels).
- You can map return values downstream. This mapping functionality is the same as for any other invoke mappings in the integration.
- You can add header support to the payload of the SOAP Adapter or REST Adapter used in the child integration. See [Create a Co-located Integration with Header Support](#).
- You can create a parent integration using a REST Adapter invoke connection configured with the OpenAPI URL connection type to pass a binary payload to a child integration. Use binary data with payloads that are unstructured and inline (for example, application/octet-stream). The file contents are preserved, but the receiver is required to determine the file type (for example, from the file name extension). The internet media type for an arbitrary byte stream is application/octet-stream.

The local integration adapter and the existing SOAP Adapter and REST Adapter differ as follows:

- You do not need to create multiple connections (SOAP/REST) to invoke the local integration service.
- Runtime communication always uses HTTP (that is, non-SSL).

This adapter is invoked internally in Oracle Integration to perform the local service invocation.

Restrictions

Note the following local integration restrictions:

- You cannot send or receive file references between parent and child integrations, in either direction, whether using local integrations or otherwise. The stage file is scoped in the context of its own flow instance and is not shared between parent and child integrations.

This is by design. Instead, you can send or receive the stage file as an attachment instead of a direct reference.

- You cannot create a child integration with a WSDL having a nested anonymous schema. As a workaround, manually edit the WSDL and make the schema non-anonymous.
- SOAP-based child integrations:
 - A parent integration cannot send an attachment to a configured child integration.
- REST-based child integrations:
 - A parent integration cannot send an attachment to a configured child integration.
 - A parent integration cannot process an XML payload that is expected by the child integration. The child flow must be configured to receive a JSON payload only, and cannot be invoked locally using an XML payload.

Best Practices for Designing Local Invoke Calls

- Calls to schedule tasks or asynchronous integrations are always fine with a local invoke because they are not processed in the context of the current node.
- Calls from nonparallel code sections are also always fine because the calling integration block is waiting for a response. There is currently no parallel capability in synchronous integrations. Therefore, a local invoke is always fine in a synchronous flow.
- Calls from parallel code sections (parallel for-each and parallel file processing) are problematic when using a local invoke because they may use the same node for processing. In those cases, calls do not benefit from parallelism across nodes and instead are competing for resources on a single node.

Invoke a Child Integration with a Parent Integration

The following steps provide an overview of creating an application integration in which a parent integration invokes a child integration.

Note:

- If you need to move a parent integration to another instance, it is best to include that integration and all its child integrations in a package.
- If the following integrations are imported from one environment to another (having different host names), then editing the local (child) integration or editing the REST Adapter in the Adapter Endpoint Configuration Wizard leads to major changes in the mapper that may require remapping.
 - Integrations in which a child integration is invoked from a parent integration.
 - Integrations with a REST Adapter using a Swagger-based connection.

1. Create and design an application integration.
2. Add an integration action to an integration in either of the following ways:

- On the side of the canvas, click **Actions**  and drag the **Integration** action to the appropriate location.

- Click  at the location where you want to add the integration action, then select **Integration**.

The Integration Adapter Wizard is displayed.

- Specify the following details, and click **Next**.

Element	Description
What do you want to call your local integration invocation?	Specify a name.
What does this local integration invocation do?	Specify a description.

- Specify the following details, and click **Next**.

Element	Description
Select Integration Context Note: This field is only displayed if the parent integration is in a project.	Select the location of the child integration to invoke. <ul style="list-style-type: none"> Project: Select this option, then choose the specific project. Your current project is highlighted and selected by default at the top of the list. Two types of child integrations are shown for selection: <ul style="list-style-type: none"> If you select the project you are already in, all integrations are shown, regardless of whether they are set as public. If you select a different project, all integrations that are set as public in that project are shown. See Set a Project Integration as Public. <ul style="list-style-type: none"> Not in Project: Select to invoke a child integration in the global context (that is, not available in a project). Note: You can invoke an integration not set as public from a runtime tool such as Postman.
Integration	Select the child integration to invoke. Only active integrations are displayed.
Identifier	Displays the identifier of the selected integration. This value cannot be changed.
Description	Displays the description of the selected integration.

- Select the operation for the child integration to perform, and click **Next**. The operation depends on the child integration. Note the following:
 - If the child integration trigger connection is a REST Adapter, it can have Get, Post, Put, and Delete operations.
 - If the child integration trigger connection is a SOAP Adapter, it can have the WSDL operation name.
 - If the child integration trigger connection is a schedule, it processes a submit now configuration.
- Review your selections, and click **Done**.

7. Complete design of the parent integration.
8. Specify appropriate business identifiers for tracking the integration at runtime.
9. Activate and invoke the integration.
10. In the navigation pane, click **Observability**, then **Instances**.
If successful, the Instances page has two instances (one for the parent and one for the child). If unsuccessful, and the parent is unable to call the child, there is one entry (for the parent). If the parent can call the child, there are two entries. If the child fails or the child was successful, but the parent did not properly process the response, the parent may fail.

Create a Co-located Integration with Header Support

Headers are optional elements that pass extra information about your application requirements. For example, you can use the header element to specify a digital signature for password-protected services. You can configure and select the headers to send with the payload. Header selection is automatic based on the WSDL URL definition you provide on the Connections page.

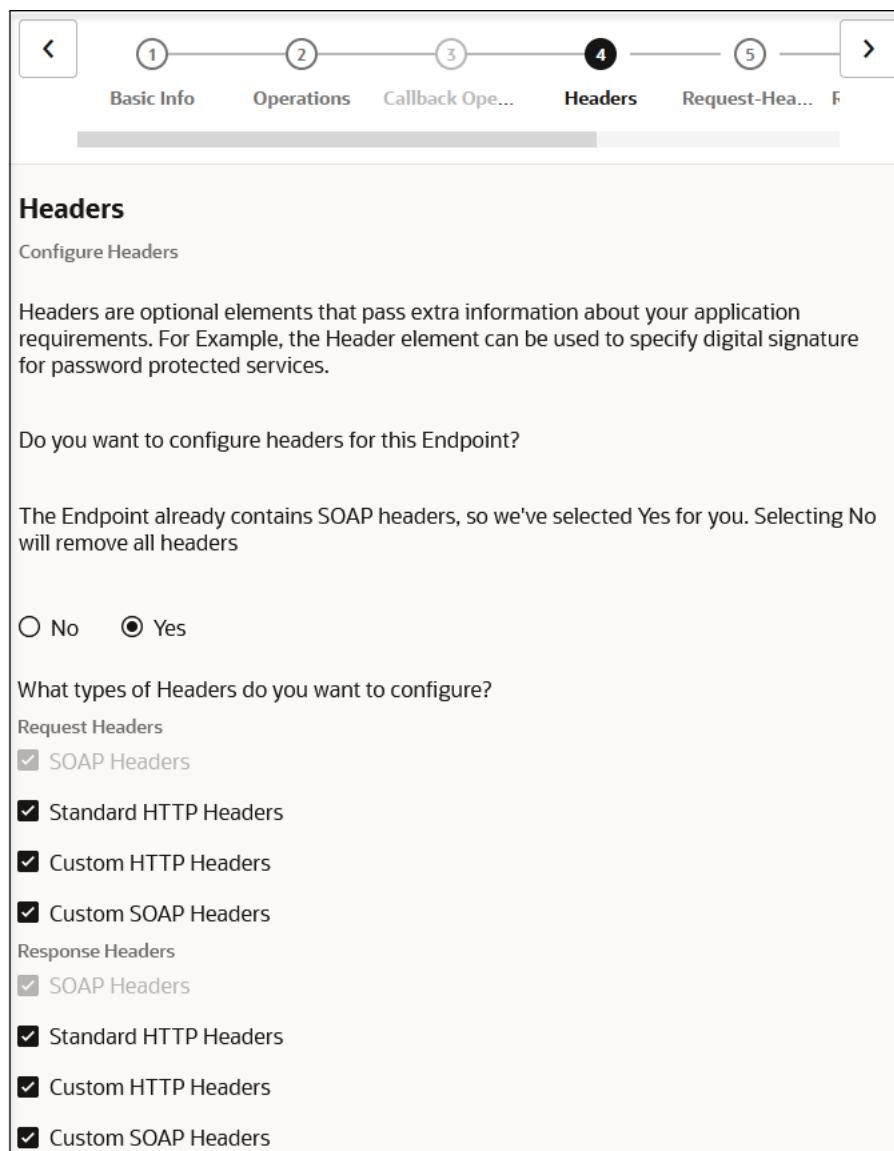
Along with request and response SOAP and REST headers, you can configure standard HTTP headers with the SOAP Adapter and REST Adapter, custom HTTP headers with the SOAP Adapter and REST Adapter, and custom SOAP headers. You can view and edit these headers in the request and response mapper under the header elements for either the SOAP Adapter or REST Adapter.

The following use case provides a high-level overview of creating a co-located integration with header support.

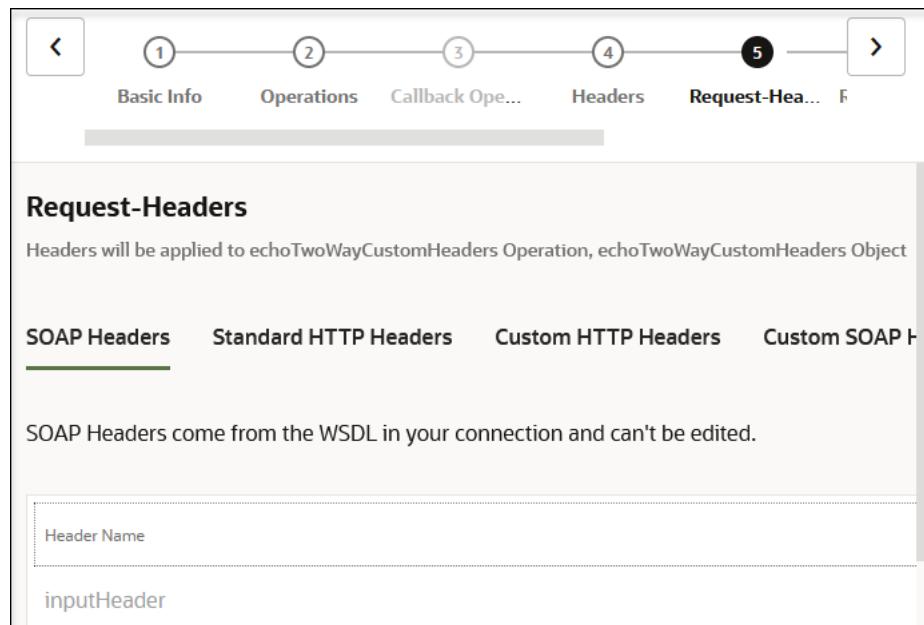
1. Create a SOAP Adapter connection (for this example) with a WSDL URL that includes the header definitions.



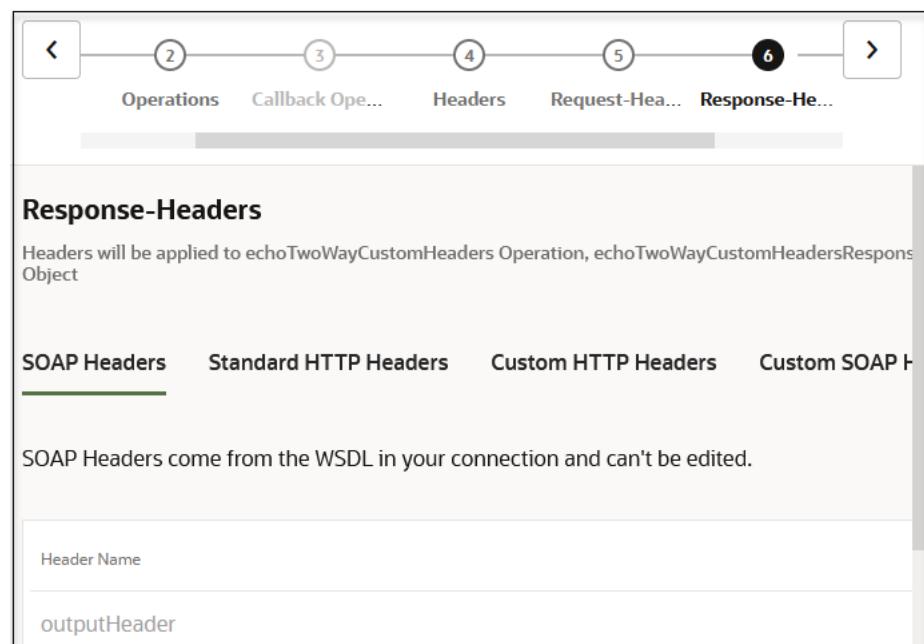
2. Complete the remaining fields on the Connections page.
3. Test and save the connection.
4. Create an integration to act as the co-located (child) integration.
5. Add the SOAP Adapter as a trigger connection in the integration.
6. Complete the pages of the Adapter Endpoint Configuration Wizard. In particular:
 - a. On the Headers page, specify request and response header details.



- b. On the Request-Headers page, specify request header names.



- c. On the Response-Headers page, specify response header names.



7. In the mapper, map the request and response headers. While this example maps headers between two SOAP Adapters, you can also map request and response headers between two REST Adapters.
8. Save and activate the integration.
9. Create another integration to act as the parent integration.
10. Add the REST Adapter as a trigger connection in the integration.
11. Configure the pages in the Adapter Endpoint Configuration Wizard.
12. Add an **Integration** action to the integration.

This invokes the Integration Adapter Wizard.

13. Complete the pages of the wizard. In particular, select the child integration created in Step 4 on the Selection Integration page.
14. Click the map icon between the REST Adapter trigger connection and SOAP invoke (the **Integration** action you configured) to configure the map.
15. Map the source **id** element to the target SOAP, HTTP, and custom header elements.
16. Save and activate the parent integration. The parent integration is now designed with a child integration as a SOAP invoke.

Backward Compatibility for Header Support

Note the following details when using header support in existing integrations created prior to the release of support for headers in co-located integrations.

Question	Answer
What happens if an existing integration created with a header is imported into an environment in which header support in co-located integrations is supported.	You can activate the integration without any issues. However, if you want to use header functionality, you must edit the adapter by clicking through the pages of the Adapter Endpoint Configuration Wizard, then clicking Done on the Summary page. In this scenario, mapping is not deleted; you must correct it.
What happens if you perform the following steps. <ol style="list-style-type: none">1. Create a child flow with headers in a trigger connection.2. Use this child endpoint as a SOAP connection.3. Use the above SOAP connection as a trigger and create an integration in which you set Do you want to configure headers for this Endpoint? to No on the Headers page of the Adapter Endpoint Configuration Wizard.4. Activate the integration.5. Invoke this child integration in a parent integration that uses the Integration action.6. Check if the integration generates a wrapper in the local invoke.	Based on the selection of Yes or No , headers are visible or invisible in the mapping.

Dynamically Invoke a Child Integration

You can dynamically invoke a (co-located) child integration at runtime. In this example, a parent integration can call either of two child integrations at runtime based on the data. The parent integration is designed to invoke child integration one. However, it can also invoke child integration two. The code and version values are passed to the appropriate child integration.

 **Note:**

Dynamic invocations are only supported with the REST Adapter.

1. Create and activate an initial child integration with a REST Adapter trigger connection.
 - a. Create an integration (for this example, named `childone`).
 - b. Add a REST Adapter trigger connection.
This invokes the Adapter Endpoint Configuration Wizard.
 - c. Provide the necessary details on the pages of the wizard.
 - d. Review your configurations on the Summary page, then click **Done**. For this example, the following values are configured:
 - The relative resource URI is **/emp**.
 - The verb (method) is **GET**.
 - The response media is **JSON**.

Endpoint Summary

REST Service URI:
`/emp`

Method:
`GET`

Response Media Type:
`JSON`

- e. Perform mappings in the mapper.
- f. Activate the integration.
2. Create and activate a second child integration with a REST Adapter trigger connection.
 - a. Create a second integration (for this example, named `childtwo`).
 - b. Add a second REST Adapter trigger connection.
 - c. Configure the second REST Adapter with the same values as the first REST Adapter.
 - d. Perform mappings in the mapper.
 - e. Activate the integration.
3. Create and activate a parent integration.
 - a. Create an integration.
 - b. Add a REST Adapter trigger connection.
 - c. Provide the necessary details on the pages of the wizard, including creating the `code` and `version` query parameters on the Request Parameters page.
 - d. Review your configurations on the Summary page, then click **Done**. For this example, the following values are configured:
 - The relative resource URI is **/employeeinfo**.
 - The verb (method) is **GET**.
 - The query parameters are **code** and **version**.
 - The response media is **JSON**.

Endpoint Summary	
REST Service URI:	/employeeinfo
Method:	GET
Query Params:	code, version
Response Media Type:	JSON

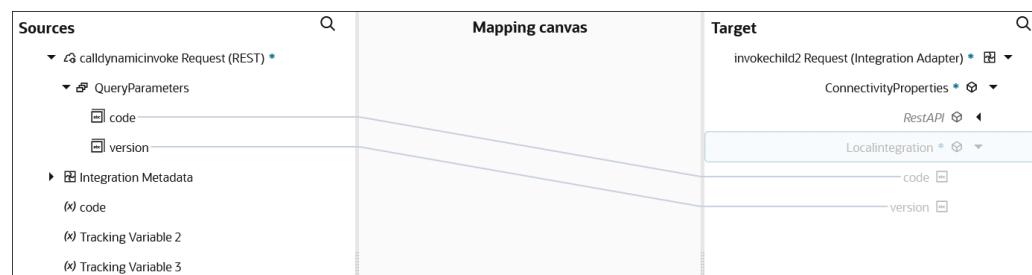
- e. Add an **Integration** action to the integration.

This invokes the Integration Adapter Wizard.

- f. Select the child integration to invoke. For this example, **childdone** is selected.
- g. Select the operation.

The screenshot shows the Integration Adapter Wizard interface. At the top, there is a navigation bar with four steps: 1. Basic Info, 2. Select Integration, 3. Select Operation (which is highlighted with a black circle), and 4. Summary. Below the navigation bar is a section titled "Select Operation". It has a label "Operation *". Underneath it is a "Filter by Operation" input field containing the text "getEmployee".

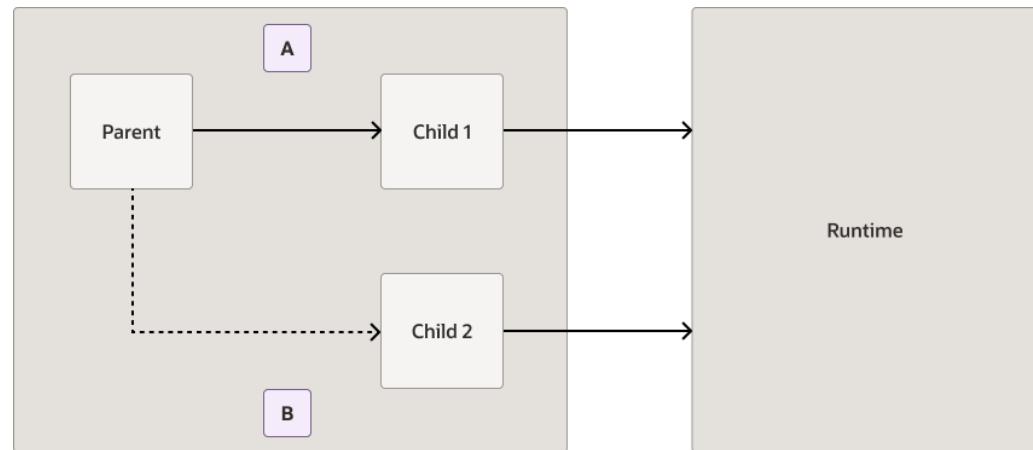
- h. Complete the wizard, and click **Done**.
- i. Map the **code** and **version** source query parameters to the **code** and **version** target local integration elements.



- j. Complete design of any remaining actions in the integration.
- k. Activate the integration.

Even though the parent integration was designed to invoke only child integration one, you can now invoke either child integration by replacing the code and version of that child integration in the endpoint below.

```
http://host_name:port/ic/api/integration/v1/flows/rest/  
DYNAMIC_INVOKE/1.0/employeeinfo?  
code=[code-value]&version=[version-value]
```



Where:

- A - `http://host_name:port/ic/api/integration/v1/flows/rest/DYNAMIC_INVOKE/1.0/employeeinfo? code=CHILDONE&version=1.0`
- B - `http://host_name:port/ic/api/integration/v1/flows/rest/DYNAMIC_INVOKE/1.0/employeeinfo? code=CHILDTWO&version=2.0`

Manage Packages

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

Topics:

- [View the Integrations in a Package](#)
- [Import a Package](#)
- [Export a Package](#)
- [Update the Package for an Integration](#)
- [Delete a Package](#)
- [Convert a Package to a Project](#)

When you create an integration, you can also create a package or select an existing package in which to include the integration. See [Create an Integration](#).

View the Integrations in a Package

You can view the integrations included in a package.

1. In the navigation pane, click **Design**, then **Packages**.

The Packages page is displayed. The package names and the number of integrations included in each package are displayed. If you have not yet created or imported a package into Oracle Integration, this page is empty.

A screenshot of the Oracle Integration Packages page. The title bar says "Packages". Below it is a search bar with a magnifying glass icon and a filter icon. On the right side of the header are "Import" and "C" buttons. The main area shows a table titled "150 Packages". The columns are "Name", "Type", and "Integrations". The data rows are: A125 (Developed, 1), AngTest (Developed, 1), AutoPackforOutofQMS (Developed, 1), Back_G3_MIG (Developed, 1), cbr.mep04.flow.functional.test.series.2 (Developed, 1), com.lmg.misc (Developed, 2), com.lmg.transactions (Developed, 8), com.mathworks.erp.otc (Developed, 1), and ERPCloud_Patterns (Developed, 5).

2. Use **Search** or **Filter** to find the package you want, then click the package name.
The integrations included in the selected package and their current states are displayed (for example, active or inactive).
3. Click **Close**.

Import a Package

You can import a package of integrations into Oracle Integration 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 Cloud Marketplace.

Note:

You cannot import a package if it contains activated integrations. Ensure that all integrations are deactivated before including them in a package archive (PAR) file.

1. In the navigation pane, click **Design**, then **Packages**.
2. In the banner, click **Import**.
3. Browse for and select the PAR file when prompted.
4. Click **Import**.

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

You can import prebuilt packages from Oracle Cloud Marketplace. See [Import a Prebuilt Integration](#) and Get Familiar with the Home Page in *Getting Started with Oracle Integration 3*.

Export a Package

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

1. In the navigation pane, click **Design**, then **Packages**.
2. Hover over the package to export.
3. Click **Actions ...**, then select **Export**.
4. Click **Export** when prompted to confirm your selection.
5. 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.

Update the Package for an Integration

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. In the navigation pane, click **Design**, then **Integrations**.

2. Hover over the integration of the package that you want to update, then click **Edit** .

 **Note:**

The integration must not be active.

3. Click **Primary Info** .
4. In the **Package** field, enter a new package name or search for an existing package name (as you type the initial letters, the existing package is displayed) to move your integration to an existing package.
5. Click **Save**.
6. In the navigation pane, click **Packages**.
7. Click the package name you specified to see your integration.

Delete a Package

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

1. In the navigation pane, click **Design**, then **Packages**.
2. Hover over the package to delete.
3. Click **Actions ...**, then select **Delete**.
4. Click **Delete** when prompted to confirm your selection. The package and *all* of its integrations are deleted.

 **Note:**

If you want to delete a package, but not the integrations, navigate to each integration listed in the package and clear the **Package** field or replace the package name with a different name. When the last integration is removed from the package, the package is automatically deleted. See [Update the Package for an Integration](#).

Convert a Package to a Project

You can convert a user-developed package into a project in Oracle Integration. This action creates a project that includes all resources in the package (integrations, connections, lookups, and JavaScript libraries). The existing package remains untouched.

1. In the navigation pane, click **Design**, then **Packages**.
2. Hover over the user-developed package to convert to a project. Accelerator and recipe packages cannot be converted to projects.
3. Click **Actions ...**, then select **Create Project**.
4. Enter a name and optional keywords and descriptions for the project.
5. Review the following options to determine if you want to select the **Anyone can edit, view, and monitor** check box.
 - Select this option to allow anyone to edit, view, and monitor resources in the project. You can always restrict access at a later time.
 - Don't select this option if you want to restrict access to editing, viewing, and monitoring resources in this project. After project creation, you can select the specific groups and users to which to provide access to this project. See [Control Who Can Edit, View, and Monitor in a Project](#).
6. Click **Create**.

A message is displayed.

Project *project_name* has been submitted for creation. Click **here** to see the newly created project.

7. Click the link to access the Projects page.
8. Open the project and note that the resources available in the package (integrations, connections, lookups, and JavaScript libraries) are now available in the project.

Import and Export Integrations

You can import and export integrations to share them between Oracle Integration environments.

See the following topics:

- [Import an Integration](#)
- [Import a Prebuilt Integration](#)
- [Export an Integration](#)

Import an Integration

You can import integrations that were previously exported as an IAR file from Oracle Integration. You must update your connection properties after the import.

Note:

- You cannot manually edit an exported integration (IAR) file outside of Oracle Integration and then import it into another instance. For example, if you manually update a connection name in an exported integration, then import it, the connection created cannot be updated or deleted. If you need to modify the file, contact Oracle Support Services.
- You cannot import an Oracle Integration 3 integration into Oracle Integration Generation 2.
- If you import a basic routing integration into Oracle Integration 3, it is converted to an application integration.

To import an integration:

1. In the navigation pane, click **Design**, then **Integrations**.

2. In the banner, click **Import**.

The Import integration panel opens.

3. Click **Drag and Drop** to select a file or drag one to this field.

4. Select one of the following options.

- **Import:** Imports the integration for you to configure your connection properties at a later time.
- **Import and Configure:** Imports the integration and shows the Configuration Editor page for configuring your connection properties now.

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 for you to configure or activate.

Import a Prebuilt Integration

You can import prebuilt integrations into your Oracle Integration environment.

There are two types of prebuilt integrations:

- User-created integrations. These are integrations that you or another user created. See [Import an Integration](#).
- Oracle-created integrations from the Oracle Integration Home page or from the Oracle Cloud Marketplace as part of a package. Integrations from the Home page are designated with a **RECIPE** message that is displayed next to the integration name on the Integrations page. See Get Familiar with the Home Page in *Getting Started with Oracle Integration 3* and the [Oracle Cloud Marketplace](#).

Export an Integration

Once you create an integration, you can export that integration as an IAR file for use in other Oracle Integration environments. You can export a locked integration. You can also export integrations that publish and subscribe to events and import them into a project.

To export an integration:

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the integration to export.
3. Click **Actions . . .**, then select **Export**.

A message is displayed indicating that the integration has been downloaded. The file is saved with a name that consists of the identifier plus the version number, and an IAR extension.

You can import the exported integration into the XSL Map Editor in Oracle Service Bus. See [Import a Map File into Oracle JDeveloper](#).

 **Note:**

Lookups referenced using the `lookupValue` function in the Expression Builder are included in the exported integration JAR file. When you import the integration, the referenced lookups are also imported and are visible in the Expression Builder.

Clone the Design-Time Metadata of an Entire Service Instance

You can clone (create) archives of integration design-time metadata to export and import between Oracle Integration 3 instances. This feature can be useful if you want to take a snapshot of a production environment and try to recreate a problem in a different environment, move metadata from an instance in one region to an instance in another region, perform manual archives, or automate your environment to archive daily to a repository such as Git.

 **Note:**

You cannot export Oracle Integration Generation 2 archives and import them into Oracle Integration 3 or export Oracle Integration 3 archives and import them into Oracle Integration Generation 2.

Topics:

- [Step 1: Create an Object Storage Bucket and Construct the Storage URL](#)
- [Step 2: Configure the Instance Object Storage Bucket](#)
- [Step 3: Export and Import Design-Time Metadata Between Instances](#)

Step 1: Create an Object Storage Bucket and Construct the Storage URL

You must create an object storage bucket (if one does not exist) and construct the storage URL in the Oracle Cloud Infrastructure Console in order to export and import design-time metadata

into an Oracle Integration instance. You specify this URL during object storage bucket configuration on the Instance storage page.

1. Log in to the Oracle Cloud Infrastructure Console with the user account that your administrator added to the group for exporting design-time metadata. You must have the admin role to create an object storage bucket. See *Manage Access and Assign Roles* in *Provisioning and Administering Oracle Integration 3*.
2. In the navigation pane, select **Storage**.
3. Click **Buckets**.
4. Select the compartment in which to create the storage bucket.
5. Click **Create Bucket**.

Buckets in OIC				Compartment
Object Storage provides unlimited, high-performance, durable, and secure data storage. Data is uploaded as objects that are stored in buckets. Learn more				
Name	Default Storage Tier	Visibility	Created	⋮
bulk-orders	Standard	Private	Thu, Apr 27, 2023, 16:33:03 UTC	⋮
bucket-demo	Standard	Private	Mon, Mar 13, 2023, 08:14:45 UTC	⋮

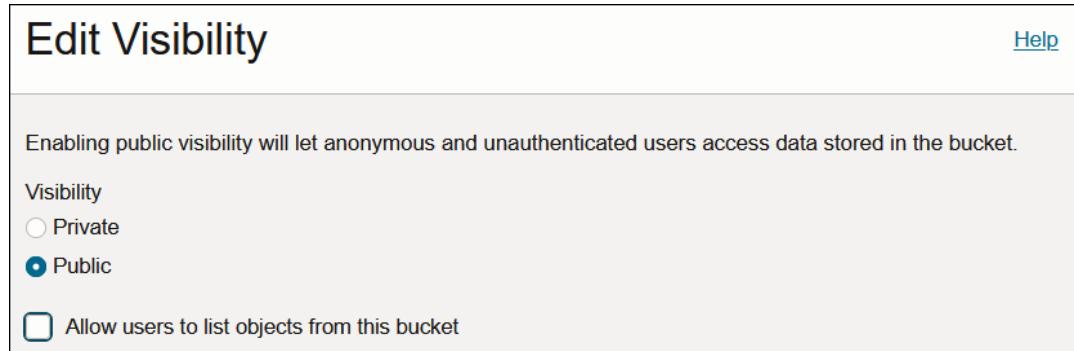
6. Enter the following information, then click **Create**.

Element	Description
Bucket Name	Enter a name.
Storage Tier	Select Standard .
Enable Auto-Tiering	Automatically move infrequently accessed objects from the Standard tier to less expensive storage. See Managing Buckets .
Enable Object Versioning	Create an object version when a new object is uploaded, an existing object is overwritten, or an object is deleted. See Using Object Versioning .
Emit Object Events	Create automation based on object state changes using the Events Service. See Overview of Events .
Uncommitted Multipart Uploads Cleanup	Create a life cycle rule to automatically delete uncommitted multipart uploads older than seven days. See Using Object Lifecycle Management .
Encryption	<ul style="list-style-type: none"> • Encrypt using Oracle managed keys Leaves all encryption-related matters to Oracle. • Encrypt using customer-managed keys Requires a valid key from a vault that you can access. See Overview.
Resource logging	Click to enable resource logging to allow resource tracking, troubleshooting, and data insights. By default, resource logging is disabled.
Tags	Add tags to organize your resources.

The new bucket is displayed in the list.

7. To view bucket details, click the name.
8. Click **Edit Visibility**.

9. Click **Public** and deselect the **Allow users to list objects from this bucket** check box. This action ensures that the storage bucket you create is publicly visible. Otherwise, the export utility cannot see the storage.



10. In the navigation pane, select **Identity & Security**, then **Users**.

11. Select the appropriate user.

12. Under **Resources**, select **Auth Tokens**.

13. Click **Generate Token**.

14. Provide a token description and click **Generate Token**.

15. Copy and save the generated token.

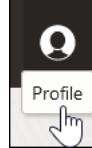
The storage payload format is as follows with object storage bucket details:

```
{"storageInfo": {
    "storageUrl": "https://swiftobjectstorage.region.oraclecloud.com/v1/
namespace/bucket",
    "storageUser": "my.email@oracle.com",
    "storagePassword": "generated_token"
}}
```

For example:

```
{
  "storageInfo": {
    "storageUrl": "https://swiftobjectstorage.us-
ashburn-1.oraclecloud.
com/v1/paasdevoic/cloneRepo",
    "storageUser": "my.email@oracle.com",
    "storagePassword": "generated_token"
  }
}
```

16. Construct the storage URL using the following example. You must specify the URL later when configuring the object storage bucket. See [Step 2: Configure the Instance Object Storage Bucket](#).

Step	Description	Example
1	Enter the Swift API as the base for the URL because you are dealing with object storage.	<code>https://swiftobjectstorage.us-</code>
2	Enter a required data center. To find a data center, go to the drop-down menu in the banner (for example: us-ashburn-1).	<code>https://swiftobjectstorage.us-ashburn-1</code>
3	Enter the domain name. This is typically oraclecloud.com.	<code>https://swiftobjectstorage.us-ashburn-1.oraclecloud.com</code>
4	Enter the version: v1.	<code>https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1</code>
5	Enter the automatically generated object storage namespace. a. In the upper right, click the Profile menu.	<code>paasdevoic</code>
		
	b. Click Tenancy: your_tenancy_name . The namespace string is listed under Object Storage Settings . See Understanding Object Storage Namespaces .	
6	Enter the bucket name you created in Step 6 as the last part of the URL.	<code>https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/paasdevoic/BUCKET_NAME</code>

Examples of the full storage URL to use later on the Instance storage page in [Step 2: Configure the Instance Object Storage Bucket](#) are as follows:

`https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/paasdevoic/cloneRepo`

`https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/paasdevoic/migrationartifacts_bucket-20190717-1223`

Step 2: Configure the Instance Object Storage Bucket

You must specify the Swift URL location and user name and password credentials of your existing object storage bucket instance before you can create export and import archives of design-time metadata on the Import and Export page.

To complete the fields on this page, an object storage bucket instance must already exist. Otherwise, you must create a new instance. See [Step 1: Create an Object Storage Bucket and Construct the Storage URL](#).

1. In the navigation pane, click **Settings**, then **Storage**.
2. Enter the following details. You specified these details when creating your object storage bucket. See [Step 1: Create an Object Storage Bucket and Construct the Storage URL](#).

Element	Description
Name	Enter the name of the object storage bucket.
Swift URL	<p>Enter the object storage bucket Swift URL you constructed.</p> <p>For example:</p> <pre>https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/paasdevoic/cloneRepo</pre> <p>See Step 1: Create an Object Storage Bucket and Construct the Storage URL.</p>
User	<p>Specify the object storage bucket user name. The user name and password were created when you generated the token in the Oracle Cloud Infrastructure Console. See Step 1: Create an Object Storage Bucket and Construct the Storage URL.</p> <p>Note: Ensure that the user name is prefixed with the domain name. For example:</p> <pre>primordialidcsprovider/clone-test-user</pre> <p>You can also get this user name from your profile link in the top right corner of the Oracle Cloud Infrastructure Console.</p>
Password	Specify the password.

Identify the object storage bucket. This object storage will be used to store and retrieve exported archives.

Name
DEV_CLONE_STORAGE

Swift URL
https://swiftobjectstorage.oraclecloud.com/v1/paasdevoic/clo

Username
primordialidcsprovider/clone-test-user

Password
.....

- Click **Save**. You can now export and import archives of design-time metadata on the Import and Export page.

Step 3: Export and Import Design-Time Metadata Between Instances

You can export and import archives of integration design-time metadata between instances. This feature can be useful if you want to take a snapshot of a production environment and try to recreate a problem in a different environment, move metadata from an instance in one region to an instance in another region, perform manual archives, or automate your environment to archive daily to a repository such as Git.

Note:

- Ensure that you first configure the Swift URL location and username and password credentials of the object storage bucket instance to which to export an archive of your design-time metadata on the Instance storage page. See [Step 2: Configure the Instance Object Storage Bucket](#).
- You cannot export Oracle Integration Generation 2 archives and import them into Oracle Integration 3 or export Oracle Integration 3 archives and import them into Oracle Integration Generation 2.
- This feature cannot be used to export specific integrations. All integrations are exported.
- The target instance must be completely empty prior to importing the exported archive job.

- [Create an Export Job](#)
- [Create an Import Job](#)

Create an Export Job

You create an export job that consists of an archive file of design-time metadata that you want to export to the object storage bucket you configured on the Instance storage page.

- In the navigation pane, click **Settings**, then **Import and export**.

The Import and Export page is displayed with the status of any import and export jobs.

Import and Export				
<input type="button" value="Import"/> <input type="button" value="Export"/> <input type="button" value="C"/>				
3 jobs				
Name	Status	Duration	Start time	
<input checked="" type="checkbox"/> export_si01-manual-l2-log-testing-1	Completed	8 Minutes	Nov 9, 2023, 12:25:36 AM PST	
<input checked="" type="checkbox"/> export_si01-manual-l2-log-testing	Completed	6 Minutes	Nov 8, 2023, 06:43:11 AM PDT	
<input checked="" type="checkbox"/> Export of instance si01-manual-l2-development: Export of l2 manual si01	Failed	2 Minutes	Nov 3, 2023, 05:17:19 AM PDT	

2. Click **Export** to create a job. A job consists of an archive file of design-time metadata that you want to export to the object storage bucket. If you have not configured an object storage bucket, you are prompted to click **Configure now**.
3. Complete the following fields.

Element	Description
Job Name	Enter a unique job name or accept the default value.
Export security artifacts	Select the check box to export the following security artifacts with your job: <ul style="list-style-type: none"> • Security policies • Security credentials (for connections) • Customer certificates
Description	Enter an optional description that describes the export job.

Create export job

Start a job to export the current instance to an archive file in your allocated storage area.

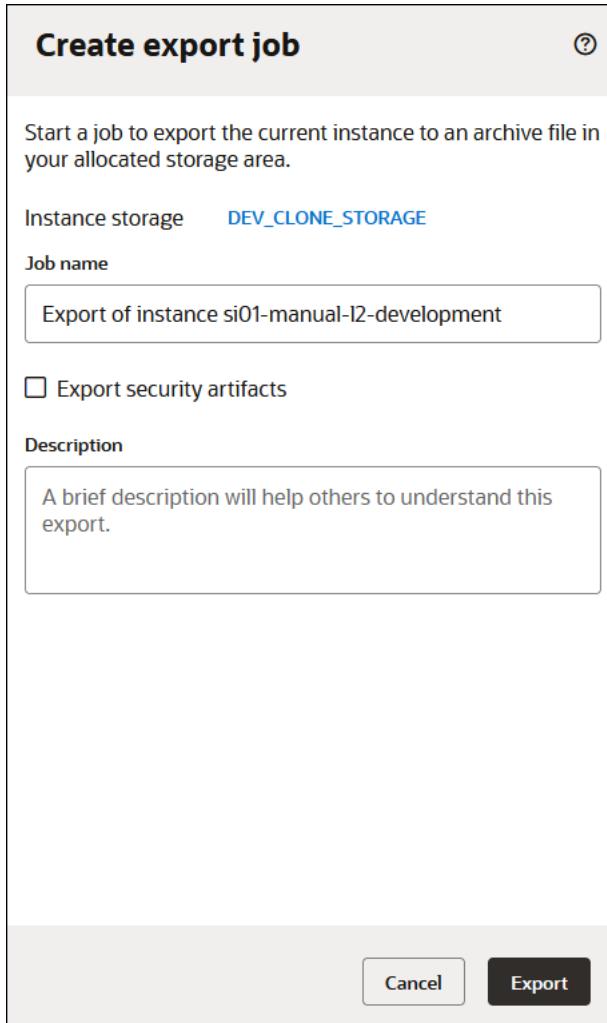
Instance storage [DEV_CLONE_STORAGE](#)

Job name

Export security artifacts

Description

[Cancel](#) [Export](#)



4. Click **Export.**

A message is displayed in the banner at the top of the page.

Export job has been successfully started.

- 5.** View the status of export job creation and click **Refresh**  periodically to view progress. When the export job completes successfully, **Completed** is displayed in the **Status** field.
- 6.** Click **Details**  to view details about the job.
- 7.** If export archive creation does not complete successfully, click **Download detailed report**  to download a report about the export job.

Create an Import Job

You create a job to import the exported archive job from the object storage bucket instance into the new instance.

- 1.** Sign in to the instance in which to import the exported archive of design-time metadata.

 **Note:**

The target instance must be completely empty prior to importing the exported archive job.

2. In the navigation pane, click **Settings**, then **Import and export**.
3. Click **Import** to create a job to import the exported archive job from the object storage bucket instance into the new instance.
4. Complete the following fields.

Element	Description
File name	Select the archive from the list to import into the instance.
Import options	Select the import mode: <ul style="list-style-type: none"> • Import: Imports all integrations in the archive. You can also select Activate and Start schedules to activate all integrations and start all schedules during this same import session or during a separate session at a later time. Selecting those options separately enables you to first update any configuration properties in the imported integrations (for example, modify any necessary configuration or security properties on the Connections page for each integration). • Activate: Activates all integrations imported with the Import option during the same session or during a separate session. You can also select Start schedules to start any integration schedules. • Start schedules: Starts integration schedules during the same session in which you selected Import and Activate or during a separate session.
Import security artifacts	Select this check box if you previously selected Export security artifacts when creating your export archive job. You can also show this option by clicking Import .
Job name	Enter a unique job name or accept the default value.
Description	Enter an optional description that describes the import job.

Create import job

Start a job to import a saved archive into the current instance from your allocated storage area.

Instance storage [DEV_CLONE_STORAGE](#)

File name

Select file name

Required

Import options

Import

Include security artifacts

Activate

Start schedules

Job name

Import of instance si01-manual-I2-development

Description

A brief description will help others to understand this import.

[Cancel](#) [Import](#)

- Click **Import** to start the job to import the archive from the object storage bucket instance into the new Oracle Integration instance.

A message is displayed in the banner at the top of the page.

Import job has been successfully started.

- View the status of import job creation and click **Refresh**  periodically to view progress. When the import job completes successfully, **Completed** is displayed in the **Status** field.
- Click **Details**  to view details about the job.
- If the import archive is not successful, click **Download detailed report**  to download a report about the import job.
- Browse the pages and note that the design-time metadata you exported is now visible. For example, for Integrations, look for integrations, connections, lookups, and more.

 **Note:**

Your archive file resides in the object storage bucket until you delete it.

Refresh Endpoints for Integrations

After you clone an integration, customize a prebuilt integration, or import an existing integration into Oracle Integration, you update the connection information (WSDL, user name, 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. For custom fields or updated fields to appear in the mapper, you must refresh the endpoints in Oracle Integration.

As an example, the WSDLs for one of your connections (for example, a Salesforce connection) change frequently and you must uptake the latest WSDLs into your integrations. By refreshing the WSDL file, the custom fields of imported mappings are not deleted, and are available for editing, when needed. This eliminates the need to remap source and target elements completely from scratch in the mapper.

Endpoint refreshing is an asynchronous operation. This reduces the chance of timeouts occurring. The refresh can scale from a single action to an entire integration.

Restrictions

- Integrations cannot be refreshed when they are active, being activated, or being deactivated.
- Integrations are locked during the refresh. No users can modify the integration during this period.
- Integration activation is blocked during a refresh.
- During endpoint refreshing, you can view or exit the integration canvas. If you are still on the canvas when endpoint refreshing completes, the canvas refreshes and you can resume editing. If you leave the canvas, you receive an alert that the refresh has completed, and the lock is released.
- The Connections page shows a warning message if you attempt to edit during integration refreshing.
- You cannot unlock an integration until refreshing completes.
- Once refreshing completes successfully, you cannot roll back to the previous state.
- Minimize your use of refreshing. Only use this feature when absolutely necessary.
- There cannot be root level differences between the old and new WSDLs. If there are differences, WSDL refreshing fails. Therefore, you can change the connection information, but must ensure that the new WSDL does not have root element differences from the previous WSDL.
- If you create an integration in which Oracle Service Cloud (RightNow) Adapter WSDL version 1.2 is used and try to refresh it with Oracle Service Cloud (RightNow) Adapter WSDL version 1.3, a mapper error occurs.
- JCA artifacts are not refreshed.
- A new WSDL can have references to a different schema with different data structures (for example, you change the schema by adding, modifying, or deleting a new complex element to the root element), *only* if the root element within the schema is *not* modified, added, or removed. In this situation, the child element within the root element can be changed or refer to any new data structure or data type.

The refresh logic follows the reference and imports or includes the new data structure to the new WSDL. The root element within the schema is the first level of <xs:element/> or

type element (`<xs:complexType/>` or `<xs:simpleType/>`). For the first level of `<xs:element/>`, the following applies:

- Their QName cannot be modified.
- A new root element cannot be added.
- The old root element cannot be removed.

For the first level of a type element, the following applies:

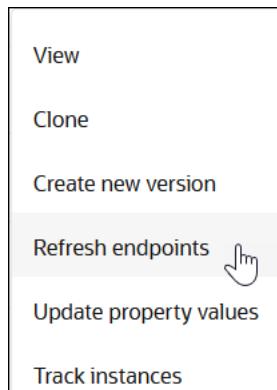
- Their QName must not change.
- The old root type must not be removed.
- A new type element can be introduced by other referencing child elements or types.

For example, one child element or type defined in a root element can refer to a new type element. The refresh logic imports or includes them into the existing schema if it has the same target namespace as the child element/type or by creating a new schema if it has a different target namespace from the child element/type. The new root level type element must be referenced by another child element/type. If it is a standalone root type element, it is not imported or included in the final refreshed WSDL.

 **Note:**

If you receive an error during the refresh, do not save the endpoint/mapping updates. Instead, close and discard the integration without saving, reopen the integration for editing, and try refreshing again.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Refresh the endpoints in either of two ways:
 - a. Hover over the name of the integration in which to refresh the endpoints.
You can refresh the endpoints for an individual endpoint or all endpoints in an integration.
 - b. To refresh the WSDL for an endpoint in the integration, click **Actions ...**, then select **Refresh endpoints** for the appropriate endpoint.



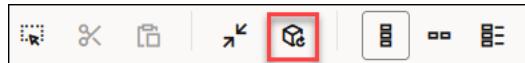
The Refresh endpoints panel opens.

Or:

- a. Click the name of the integration in which to refresh the endpoints.

The integration canvas is displayed.

- b. Click the icon in the palette above the integration to refresh the endpoints.



The Refresh endpoints panel opens.

3. Select the connections from the list for which to refresh artifacts. Refreshing endpoints may impact the business identifiers and maps.

4. Click **Refresh**.

The refresh of the impacted connection endpoints is displayed as in progress. On the Integrations page, the state of the overall integration refresh is also displayed as in progress.

This process refreshes the endpoint and any dependent artifacts to enable any custom elements to 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. If you need to cancel in-progress endpoint refreshing, click **Stop and rollback**.

See Map Data in *Using the Oracle Mapper with Oracle Integration 3* .

Set the Time Limit for Inactive Sessions

You can set the time limit for inactive sessions in an Oracle Integration instance. This setting automatically signs out a currently authenticated user after a specified period of inactivity. This setting impacts all the users of the instance and takes effect after a user signs out and signs back in.

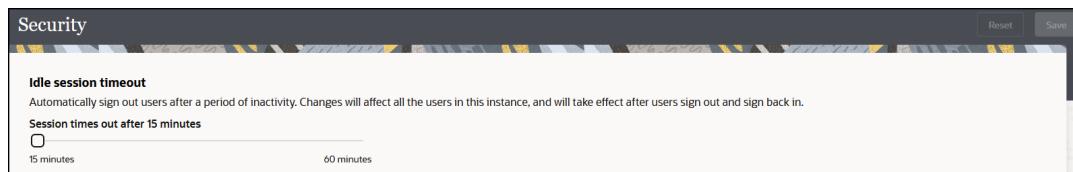
Note the following behavior:

- Only a user with the ServiceAdministrator role can view the Security page and set the time limit for an inactive session in an Oracle Integration instance.
- The default value for an inactive session in an instance is 60 minutes, unless a user with the ServiceAdministrator role sets it to another value.
- The minimum and maximum values for session inactivity timeouts are 15 minutes and 60 minutes, respectively.
- Assume you have multiple open tabs in your browser for the same integration instance.
 - You are actively working in one tab.
 - The other tab remains idle.

When the session inactivity timeout value for the integration instance is reached for the idle tab, you are not logged out of either tab.

- Assume user A is signed in when the session inactivity timeout is set to 45 minutes. A user with the ServiceAdministrator role then changes the setting to 30 minutes. The previous setting of 45 minutes remains in effect for user A as long as they are signed in. Once they sign out and sign back in, the setting of 30 minutes takes effect.

1. In the navigation pane, click **Settings**, then **Security**.



2. Drag the slider to the applicable session inactivity value.
3. Click **Save**.
4. If you want to revert to the previous session inactivity value, click **Revert**.

Modify 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. The integration must not be active. See [Deactivate an Integration](#) for instructions.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the integration to change, then click **Edit** .
3. To modify the trigger or invoke adapter connection, select the connection and click **Actions ...**, then click **Edit** .
4. Modify any necessary fields in the pages of the Adapter Endpoint Configuration Wizard.
5. To assign a new invoke connection, select the connection, and click **Actions ...**, then click **Delete** . You cannot delete a trigger connection in an integration.
6. Add the new adapter to the canvas.
7. Configure the new connection.
8. To modify a data mapping, click the appropriate map icon and update the mappings. See [Mapping Data of Using the Oracle Mapper with Oracle Integration 3](#).
9. When you are done making changes, click **Save**.

Note:

If your integration is locked and stuck in Draft mode, see [Recover Unsaved Integration Changes](#).

Impact on the Mapper of Editing the Endpoint Information in an Integration

You can edit the endpoint information in an integration that is not active. The changes that you make can impact your mappings. For example, minor edits such as changing the endpoint description do not delete the existing mappings. Major edits such as changing the selected

business objects or operations delete the mappings. In these cases, you must recreate your mappings. Before you save your updates, you are prompted to confirm your changes.

The impact of major and minor endpoint changes on an integration are as follows:

- If a minor change is detected, for example:
 - If a map is using either the request or response of the application as a primary input, output, or secondary input, the map is validated.
 - If a map is using a fault of the application as a primary input or output, the map is deleted.
 - If a map is only using a fault of the application as a secondary input, the secondary input is removed.
- If a major change is detected, for example:
 - If a map is using a request, response, or fault of the application as a primary input or output, the map is deleted.
 - If a map is only using a request, response, or fault of the application as a secondary input, the secondary input is removed.

The following are examples of major endpoint changes:

- If the application message exchange pattern changes (for example, from synchronous to asynchronous).
- If a root element name or root element namespace of the input request changes.
- If a root element name or root element namespace of the output response changes.

If none of the above changes occur, then the change is considered minor.

1. In the navigation pane, click **Design**, then **Integrations**.
 2. Hover your cursor over the integration to edit, then click **Edit** .
- The integration must not be active.
3. In the integration canvas, select the trigger or invoke endpoint to edit, then click **Edit** .
 4. Make appropriate changes in the Adapter Endpoint Configuration Wizard, then click **Done**.
 5. Select to confirm your changes when prompted. Minor edits do not delete your mappings. Major edits delete your mappings.

View the Actions and Connections in an Integration

You can view the total number of actions and invoke connections in an integration.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Click the integration to view.
3. Click **Primary Info** .
4. Click **Actions report**.
A summary of each action and invoke connection in the integration is displayed.

Properties		Actions report	
Name	Count		
Assign	20		
For each	3		
Invoke	5		
Logger	9		
Map	7		
Schedule	1		
Stage file	6		
Switch	4		
Throw new fault	1		
Total	56		

Clone 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. You can also clone a locked integration.

 **Note:**

Integration versions follow a formatting convention of `xx.yy.zzzz`, where `xx` is the major version and `yy.zzzz` is the minor version. If you clone an integration (for example, version `1.00.0000`) and change the minor version of the cloned integration to `1.10.0000`, version `1.00.0000` is deactivated when you activate version `1.10.0000`. To keep both integrations active, change the major version of the cloned integration to `2.00.0000`. This enables integration versions `1.00.0000` and `2.00.0000` to be active at the same time. See [Create an Integration](#).

1. In the navigation pane, click **Design**, then **Integrations**.
 2. Hover over the integration to clone.
 3. Click **Actions ...**, then select **Clone**.
- The Clone integration panel opens.
4. Enter a name, unique identifier, version number, documentation URL, keywords, 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`.

5. Click **Clone**.
6. You can modify the clone in any of the ways described in [Modify an Integration](#).

Create a Draft of an Integration

Creating a draft of an integration creates a new copy with the same integration information. This action represents an easier way to create an integration. During draft creation, you must update the version of the integration and optionally update the package and description. Unlike a cloned integration, you *cannot* update the integration name or identifier. You can also create a draft of a locked integration.

You cannot create a draft of a recipe or accelerator.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the integration for which to create a draft.
3. Click **Actions ...**, then select **Create new version**.

The Create new version panel opens.

4. Modify the version number and optionally the documentation URL, keyword, package, and description. Not modifying the version number results in the following error:

The version of Integration "*integration_name*" you are trying to create already exists in the system.

5. Click **Create version**.

Delete an Integration

You can delete an integration that is no longer required.

Make sure the integration you want to delete is not active. To deactivate the integration, see [Deactivating an Integration](#).

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the integration to delete.
3. Click **Actions ...**, then select **Delete**.
4. Click **Delete** when prompted.

Recipe and accelerator integrations cannot be deleted individually. You must delete the package containing these integrations.

Recover Unsaved Integration Changes

You can recover unsaved changes in an integration that crashes during design time because of a browser crash, loss of network connectivity, a server going down, and so on.

When you design an integration, a backup copy is created whenever you complete a task such as configuring an action, deleting an invoke connection, configuring business identifier tracking, or defining global faults. There is no time limit in which you need to log back in and recover your changes. The unsaved changes remain as long as the integration is not unlocked by an administrator.

The following data in an integration cannot be recovered after a crash:

- Changes made while editing an action (for example, you are designing a stage file action when your integration crashes)
- Changes made while working inside the mapper
- Layout changes such as repositioning an action
- Connections page edits

Recovery behavior is different based on the roles assigned to the user. For example, assume you are designing the following integration when your browser crashes.

1. Design an integration as a user with the ServiceDeveloper role. Assume you have not saved your changes when your browser crashes.
2. If you log back in as that same user with the ServiceDeveloper role, perform the following tasks to recover your unsaved changes.
 - a. Go to the Integrations page and note that the words **Unsaved Changes** appear next to the impacted integration.
 - b. Select to resume editing and recover your unsaved integration changes in any of the following ways:
 - Click the integration name in the **Name** field.
 - Click **Edit** .
 - Select **Resume Edit** from the **Actions ...** menu.
 - Select **Unlock** from the **Actions ...** menu, then select **Edit & Recover**.

Your integration design is displayed. The following message appears in the banner.

Unsaved changes for integration *integration_name* (1.0) have been recovered and are displayed in the canvas, but not saved yet.

- c. Click **Save** to save your changes and continue with integration design.
3. If you instead log in as a user with the ServiceAdministrator role, note the following behavior.
 - a. Go to the Integrations page and note that the word **New** instead of **Unsaved Changes** appears next to the impacted integration.
 - b. Select from the following options. A user with the ServiceAdministrator role cannot recover the unsaved changes. Only the actual designer of the impacted integration can perform that task.
 - If you click the integration name in the **Name** field, you see none of the updates you made prior to the crash. The following message is displayed in the banner.

Edit is not possible for OpportunitytoOrder (1.0) because this integration is locked. Integration is displayed in view mode.

- If you select **Edit** from the **Actions ...** menu, an **Integration Is Locked** message is displayed.
 - If you then select **Unlock** from the **Actions ...** menu, you are prompted with the **Unlock Integration** message. If you select **Unlock & Discard**, all changes are lost. Contact the integration designer to determine how to proceed.

Override Design-Time Properties in an Integration

You can configure integration properties outside of the regular integration design. At runtime, these integration properties take effect and override the values you configured at design time. For example, you can specify an email address to use in the **To** field of a notification action during runtime that overrides the email address specified at design-time. This feature enables you to configure the value without the need to edit the integration design or pass the value at runtime.

You can configure all types of properties, including file path, database properties, server properties, credentials, and any constant property that changes at regular intervals.

When you can edit the integration properties is based on the type of integration pattern you are using.

Integration Pattern	Editing Behavior
Schedule	Runtime configuration is supported. You can update the integration property values when the integration is active. When the integration is invoked: <ul style="list-style-type: none">• If the integration property value is not updated, the default value is used.• If the integration property value is updated with a new value, the new value becomes the current value and is used.
Application	Runtime configuration is not supported. You can update the integration property values when the integration is <i>not</i> active. The configured property values takes effect after integration activation. <ul style="list-style-type: none">• If the integration property value is not updated, the default value is used.• If the integration property value is updated with a new value, the new value becomes the current value and is used.

Note the following behavior regarding integration properties and major versions (for example, 01.00.0000) and minor versions (for example, 01.10.0000) of an integration.

- The current integration property values are shared by all minor version integrations with the same major version and identifier.
- Updating the current value for an integration property is reflected in all other minor version integrations if they have an integration property with the same name.
- Deleting an integration property from an integration does not remove its current value if there are other minor version integrations with the same major version and identifier.

See [Create an Integration](#) for information about versions and identifiers.

1. Click **Global Variables** (x).

2. Click the **Integration Properties** tab.

3. Click + to add a name, description, and value for the property. For example, enter an email address as the value. You can add a maximum of 10 integration properties.

The screenshot shows the 'Global Variables' section of the Oracle Integration Designer. A specific integration property, 'integration_property_1', is selected. The property details are as follows:

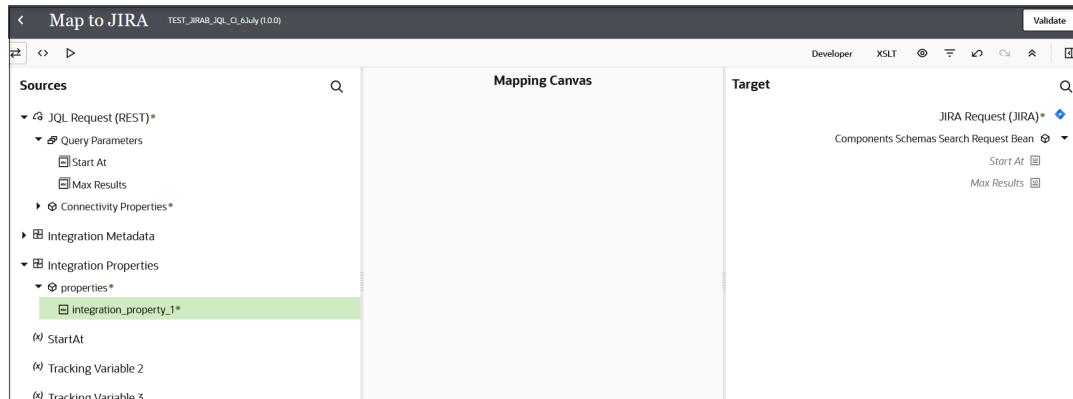
- Parameter Name ***: integration_property_1
- Description**: (empty)
- Value ***: "myemail@mycompany.com"

- When complete, click **Save** to save your changes and return to the integration canvas.

Note:

The integration property value can't be updated within the integration. For example, you cannot update the integration property in an assign action.

- In the mapper, map the integration property (named **integration_property_1** for this example) visible under **Integration Properties** in the **Sources** element tree to an appropriate element in the **Target** tree.



- Go to the Integrations page.
 - Hover over the integration whose properties you want to update.
 - Click *, then select **Update property values**.
- The Update property values panel open on the right side of the page.
- Click the integration property to update, then enter a value in the **New Value** field and click **Submit**.

Update property values
Test_ROPC_Bulk_ID_Test_2912_Delete1 (1.0)

Click to view and configure integration properties for **Test_ROPC_Bulk_ID_Test_2912_Delete1 (1.0)**. **New Value** replaces the currently stored value and will be used for the new run. **Current Value** indicates the currently stored value. Any **minor** integration having the same integration property will also refer the current value.

[Learn more](#)

integration_property_1

Property name
integration_property_1

Default value
"myemail@mycompany.com"

Current value

New value
Enter a value. Maximum 256 characters supported

[Cancel](#) [Submit](#)

A confirmation message indicates that the property was updated.

The new value is used for a new run of the integration. For this example, the email address you specified in the **New Value** field is used as the **To** field email address of the notification action in the integration.

If you select **Update property values** again to reopen the Update property values panel, the value in the **New Value** field now appears in the **Current Value** field.

You can also view previously configured integration properties.

Use Metadata in Integrations

You can access the integration name, identifier, and version instead of hard coding those values in integrations. You can also access runtime data and environment data inside the integration.

The following metadata is exposed in integrations. You can use this metadata in the mapper and actions that include the Expression Builder such as assign, log, notification, and others.

- integration
 - name

- identifier
- version
- runtime
 - instanceId
 - invokedBy
- environment
 - serviceInstanceName
 - baseURL

The following use case provides a high-level overview of using metadata in an integration.

1. Create and design an application integration.
 2. Add an action that includes an Expression Builder.
- For this example, a log action is added.
3. In the source tree, find the integration metadata.
 4. Add the required metadata to the **Logger Message** field. For this example, a **concat** function is used to include the integration **name** metadata.

5. Complete design of the integration.
6. Activate and run the integration.
7. Go to the Instances page.
8. Click the business identifier of the integration.
9. Scroll down and note that the log message in the activity stream shows the integration name (for this example, **Hello World**).

The screenshot shows the Activity Stream page with the following log entries:

- Wed Jul 06 08:38:53.161 AM IST 2022, 89ms: Wire Message received by Trigger helloworld
- 08:38:53.250 AM, 1ms: Message started processing
- 08:38:53.251 AM, 0ms: **Logger activityStreamLogger_1634894950**
 - Integration name is :Hello world
- 08:38:53.251 AM, 3ms: Data Mapping completed
- 08:38:53.254 AM, 3ms: Message sent to reply
- 08:38:53.257 AM: Reply to helloworld completed

Cancel a Running Integration Instance

You can cancel an integration instance that is not running as expected. For example, a looping action such as a while loop does not complete or an agent or cloud adapter connection does not complete.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Hover over the integration to cancel, then select **Abort**

Scheduled instances cannot be canceled from the Instances page. Selecting **Abort** for a scheduled instance causes a message to be displayed above the banner with a link to the page for canceling the running instance.

View the Contents of an Incomplete Integration

You can open an integration with incomplete connection details on the Integrations page and view its contents. These integrations can only be viewed, and not edited.

1. In the navigation pane, click **Design**, then **Integrations**.
2. Open an integration with incomplete connection details (for example, an integration imported from another instance or from the Oracle Cloud Marketplace).

3. A message is displayed at the top of the page describing why you cannot edit the integration.

Edit is not possible for *Integration_Name* (*version_number*) because the connection settings for this integration are incomplete.

Update the configuration information for the connections: *Connection_Name*. Integration is displayed in view mode.

4. Open individual adapters, mappings, actions, and other components in the integration for viewing.
5. Go to the Connections page for the adapter connection requiring configuration.
6. Update the necessary credential and security details.
7. Return to the integration and note that the message is gone.

Edit an Integration with Incomplete Connections

You can edit integrations that include endpoints based on incomplete connections.

Note the following restrictions and capabilities with editing these types of integrations:

- You can edit completed connections, actions, and maps.
 - You cannot edit endpoints based on incomplete connections. Instead, you must first edit the endpoints for those connections on the Connections page.
1. In the navigation pane, click **Design**, then **Integrations**.
 2. Open the integration that includes endpoints based on incomplete connections.
A message is displayed in the banner indicating that you can edit this integration, except for the endpoints based on incomplete connections.
 3. Edit any completed connections, actions, and maps, as necessary.
 4. To complete the integration, go to the Connections page for the incomplete connection and complete the configuration.
 5. Return to the integration and edit the newly completed connection in the Adapter Endpoint Configuration Wizard, if necessary.
 6. Save and exit the integration, then activate it.

Change the Time Zone

You can change the time zone that is displayed in Oracle Integration.

1. In the upper corner, click the **user name** icon, then select **Preferences**.
2. From the **Time Zone** 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 after 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 Oracle Integration in which the time is displayed (for example, on the Instances page).

12

Monitor Integrations During Runtime

You can monitor and manage your integrations in the runtime environment. You can also perform runtime tasks, such as tracking the status of your integration instances, viewing and diagnosing errors, and resubmitting and discarding integration failed instances.

Topics:

- [View the Dashboard](#)
- [View Notification Alert Announcements and the Status of Asynchronous Operations](#)
- [Monitor the Message Processing Status of Integrations](#)
- [Monitor Agents](#)
- [Learn About the Payloads in the Activity Stream](#)
- [Measure an Integration's Performance](#)
- [Find Integrations That Experienced Errors](#)
- [Find Integrations That Timed Out or Were Aborted](#)
- [Manage Errors](#)
- [Track Integration Instances](#)
- [Monitor Integrations That Subscribe to Events](#)
- [View Minimal Details About Debug Tracing Level Instances for 32 Days](#)
- [View the Calendar of Schedule Integration Runs](#)
- [Diagnose and Manage Event-Based Oracle Fusion Applications Integrations](#)
- [Send System Status Reports with Notification Emails](#)

View the Dashboard

You can view information about integration performance on the Integration dashboards page, including the error rate; the total number of instances, instances that succeeded, and instances in error; the number of active integrations; the status of connectivity agents; and design-time metrics.

You can view information about how your integrations are performing. The main page of the integration shows a snapshot of successful and failed integrations.

The Integration dashboards page reflects historical data. The number of all types of instance states (successful, failed, and so on) are collected hourly and displayed on the Integration dashboards page. Note the following details about the display of failed and recovered instances:

- Time 1: There are failed instances (shown in the Errors page)
- Time 2: The instances are then recovered (and are now successful)

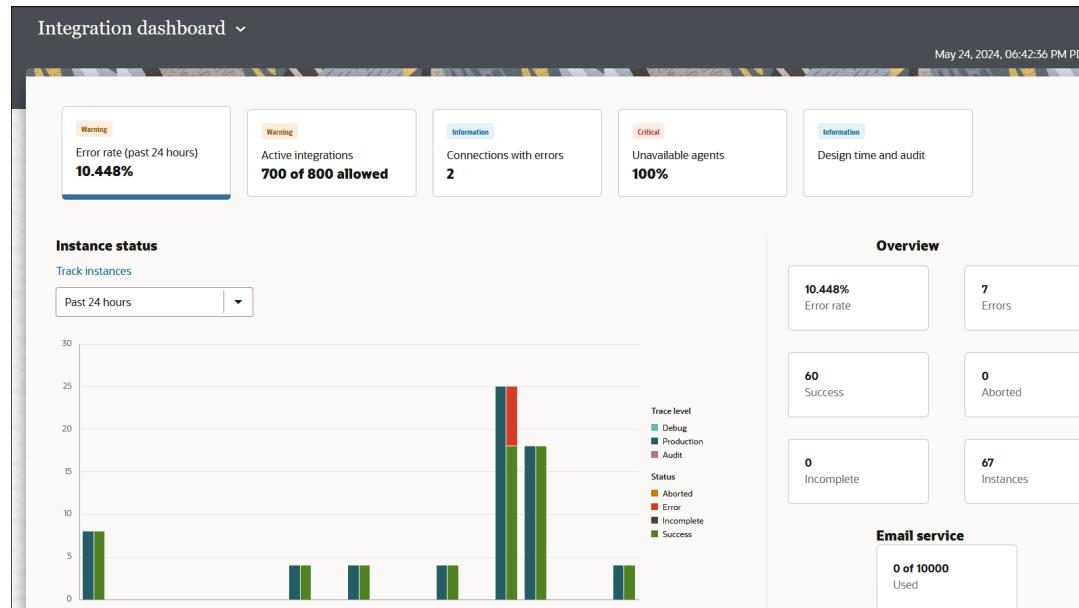
If instance state information is collected between time 1 and 2, the Integration dashboards page shows some failed instances. The history graphs on the Integration dashboards page also show the hour/day during which that was the instance state. If the recovery of failed

instances occurs before any new data is collected, the Integration dashboards page shows those instances as successful.

1. In the navigation pane, click **Observability**, then **Dashboards**.

The Integration dashboards page is displayed by default. From the drop-down list, you can also select **B2B** to view the overall status of B2B transactions. See Monitor the Overall Health of B2B Transactions in *Using B2B for Oracle Integration 3*.

2. Click a section for dashboard details.



Section	Description	Actions
Error rate (at least 24 hours) <small>(This section is displayed by default.)</small>	<p>The instance status chart shows two bars per hour.</p> <ul style="list-style-type: none"> The left bar shows the number of instances classified by their tracing level. This is important because the tracing level determines the life cycle of instances (for example, debug level tracing instances are purged in 24 hours) and the life cycle of the instance payload (for example, payloads captured for instances set to audit level tracing are purged in seven days). The right bar shows the number of instances classified by their states. The Incomplete state contains all instances that have not yet completed processing. These instances may be in the Queued, In Progress, Resubmitted, or Blocked states. <p>Data outside the gray box (usually older historical data) does not retain or contain any information about instances set to debug level tracing.</p> <p>The following graphical and numerical information is also displayed:</p> <ul style="list-style-type: none"> Error rate percentage; the number of failed, successful, terminated, and incomplete integration instances; and the total number of integration instances. Integration instances generated and processed at different tracing levels. Shows the number of emails sent per the enforced limit. The number includes successful and failed messages. See Monitor the Number of Emails Sent from Oracle Integration. 	<p>a. Select the time period for viewing integration instance status from the Instance Status list (24 hours (the default), the past week, the past month, or since the Oracle Integration instance was provisioned). The bar graphs and Overview section are refreshed to reflect information based on your selection.</p> <p>b. Click Track Instances to access the Instances page. See Track Integration Instances.</p>
Active integrations	<p>Provides statistics on the number of scheduled, application, and event integrations, including the number in projects, packages, and standalone (globally available).</p> <p>The number of active integrations per Oracle Integration instance cannot exceed 800. Active integrations are defined as currently active integrations and integrations whose activations are in progress. If this number is:</p> <ul style="list-style-type: none"> above 90% of the limit, a warning message is displayed. equal to or above the limit, a critical message is displayed. 	<p>a. Click Monitor integrations to access the Monitor integrations page. See Monitor the Message Processing Status of Integrations.</p>

Section	Description	Actions
Connections with errors	<p>Displays a graph of connections with the most errors. The connection names appear to the right of the graph. The Overview section on the right shows the type of adapters with the most errors (for example, Oracle ERP Cloud Adapter, Salesforce Adapter, and others).</p>	<p>a. Click a section in the graph to access the Errors page, which lists the integration instances that use this connection. The connection name appears in the filtered search at the top of the Errors page.</p> <p>b. Click View more under Top connections with errors to go to the Errors page. The connection names and the total instance error counts for each are displayed with the Error by connection filter automatically selected. Error count totals are displayed in descending order.</p>
Unavailable agents	<p>Displays the percentage of unavailable connectivity agents versus the number of currently running connectivity agents. The graph and Overview section provide statistics about the number of running and unavailable connectivity agents.</p>	<p>a. Click Monitor agents to access the Monitor Agents page. See Monitor Agents.</p>
Design time and audit	<p>Displays a graphical view of the state of integration components.</p> <ul style="list-style-type: none"> • Active, configured, and draft integrations • Active, configured, draft, and in-use connections • In-use and configured lookups • User-developed and recipe packages • Connectivity agent groups • Preinstalled adapters • Configured and draft libraries <p>By default, the Integrations section shows the number of integration patterns and adapters in use.</p>	<p>a. Hover over each chart to see the number of components in each state.</p> <p>b. Click the Integrations graph to show the integration patterns (application or schedule) and adapters in use.</p> <p>c. Click the Connections graph to show the number of each type of adapter in use.</p> <p>d. Click the Agents graph to show the agent groups in use.</p> <p>e. Click Audit to access the Design Time Audit page. See Check the Audit History for an Integration or Other Component.</p>

Monitor the Number of Emails Sent from Oracle Integration

You can send up to 10,000 emails in a rolling 24-hour window from Oracle Integration. However, if you use your own customer tenancy, you can increase your limit to the number allowed by the Oracle Cloud Infrastructure Email Delivery Service. When your limit is exceeded, delivery occurs successfully one last time. For the next delivery, emails stop being sent and a `Daily outbound email limit crossed` error message is written to the activity stream for the impacted integration instance.

You monitor the current email count from the **Email service** section of the Integrations dashboard page. The count includes successful and failed messages. See [View the Dashboard](#).

- [How the Rolling 24 Hour Window is Calculated](#)
- [Increase the Limit on the Number of Emails to Send](#)

How the Rolling 24 Hour Window is Calculated

- The email quota is built on a 24-hour rolling window. At any particular point in time, the last 24 hours of outbound emails are counted.
- When you exceed your quota in only one hour (same hour), you are blocked from sending emails for the next 24 hours.

To handle this type of scenario in your integration, you can also include the notification action in a scope action so that you can emit log messages in a fault handler and then resume instance processing, if required.

Increase the Limit on the Number of Emails to Send

The default method manages email delivery in Oracle Cloud Infrastructure. This tenancy limits the number of emails you can send in a rolling 24-hour window to 10,000 and the size of attachments to 2 MB.

If you instead use your own customer tenancy for sending emails, you receive the following benefits:

- The email limit is increased to the number allowed by the Oracle Cloud Infrastructure Email Delivery Service. The customer tenancy uses SMTP credentials to send email through the Oracle Cloud Infrastructure Email Delivery Service.
- You manage your approved senders and suppression list from your own customer tenancy instead of the Notifications page in Oracle Integration.

See [Configure Notification Emails](#).

Check the Audit History for an Integration or Other Component

The Design time audit page shows the audit history of integrations and other components, including administrative updates made using the File Server user interface.

Learn who updated a component, what changes they made, and when they made the update. This information is helpful for various scenarios, including troubleshooting unplanned changes. Audit history is retained for the retention period of the instance.

1. In the navigation pane, click **Observability**, then **Dashboards**.
2. Click the **Design time** section.

3. Click Audit.

The Design Time Audit page shows details about the audit for all invoked integrations.

12950 Audit Records		
Time	Action	Message
Jul 18, 2022, 10:19:51 AM MDT	unlock	User 12gen3serviceadmin unlocked Integration TEST1_ELOQU_RIGHT_CONTA_SYNC_CLO (1)
Jul 18, 2022, 10:08:05 AM MDT	lock	User 12gen3serviceadmin locked Integration TEST1_ELOQU_RIGHT_CONTA_SYNC_CLO (1)
Jul 18, 2022, 09:54:24 AM MDT	unlock	User 12gen3serviceadmin unlocked Integration TEST1_ELOQU_RIGHT_CONTA_SYNC_CLO (1)
Jul 18, 2022, 08:55:14 AM MDT	lock	User 12gen3serviceadmin locked Integration TEST1_ELOQU_RIGHT_CONTA_SYNC_CLO (1)
Jul 18, 2022, 06:17:03 AM MDT	sch_state_change	Schedule_Schedule_SFDC_CP_BULK_TEST_1507_BUL_01_00_0000 state was changed to ACTIVE
Jul 18, 2022, 06:17:02 AM MDT	sch_started	Schedule_Schedule_SFDC_CP_BULK_TEST_1507_BUL_01_00_0000 started successfully
Jul 18, 2022, 06:17:02 AM MDT	unlock	User 12gen3serviceadmin unlocked Schedule SFDC_CP_BULK_TEST_1507_BUL
Jul 18, 2022, 06:17:02 AM MDT	start_sch	User 12gen3serviceadmin submitted a request to start Schedule Schedule_SFDC_CP_BULK_TEST_1507_BUL_01_00_0000
Jul 18, 2022, 06:17:02 AM MDT	lock	User 12gen3serviceadmin locked Schedule SFDC_CP_BULK_TEST_1507_BUL
Jul 18, 2022, 06:16:53 AM MDT	unlock	User 12gen3serviceadmin unlocked Schedule SFDC_CP_BULK_TEST_1507_BUL
Jul 18, 2022, 06:16:51 AM MDT	update	User 12gen3serviceadmin updated Schedule Schedule_SFDC_CP_BULK_TEST_1507_BUL_01_00_0000

- Click **Filter** to filter data by time window, integration name, user name, or actions such as activate and deactivate. You can click the **Data range** option to specify a date range that is greater than 32 days. The maximum date range is six months. If you are using projects, the project identifier is also included in the audit logs. Use this identifier to audit instance and error details.
- Click **Download**, and select the following:
 - Download all:** Downloads a maximum of 10,000 audit records.
 - Download filtered:** Downloads records for the currently-selected filters.

For example:

```
[2022-06-18 00:57:56.372] [userId: 12gen3serviceadmin]
[12gen3serviceadmin,UNLOCK,ICS_ProjectV2,
TEST_DEMOFAULT_AGGR,TEST_DEMOFAULT_AGGR,02.00.0000] User
12gen3serviceadmin unlocked Integration
TEST_DEMOFAULT_AGGR 02.00.0000
[2022-06-18 00:38:37.468] [userId: 12gen3serviceadmin]
[12gen3serviceadmin,LOCK,ICS_ProjectV2,TEST_DEMOFAULT_AGGR,
TEST_DEMOFAULT_AGGR,02.00.0000] User 12gen3serviceadmin locked
Integration TEST_DEMOFAULT_AGGR 02.00.0000
[2022-06-18 00:37:08.655] [userId: 12gen3serviceadmin]
[12gen3serviceadmin,UNLOCK,ICS_ProjectV2,PFE_SRS_FAUL_AGGR_G3_DD_MS,
PFE_SRS_FAUL_AGGR_G3_DD_MS,02.00.0000] User 12gen3serviceadmin unlocked
Integration PFE_SRS_FAUL_AGGR_G3_DD_MS 02.00.0000
[2022-06-18 00:34:10.968] [userId: 12gen3serviceadmin]
[12gen3serviceadmin,LOCK,ICS_ProjectV2,PFE_SRS_FAUL_AGGR_G3_DD_MS,
PFE_SRS_FAUL_AGGR_G3_DD_MS,02.00.0000] User 12gen3serviceadmin locked
Integration PFE_SRS_FAUL_AGGR_G3_DD_MS 02.00.0000
[2022-06-17 23:01:14.955] [userId: 12gen3serviceadmin]
[12gen3serviceadmin,UNLOCK,ICS_ProjectV2,PFE_SRS_FAUL_AGGR_G3_DD_MS,
PFE_SRS_FAUL_AGGR_G3_DD_MS,02.00.0000] User 12gen3serviceadmin unlocked
Integration PFE_SRS_FAUL_AGGR_G3_DD_MS 02.00.0000
. . .
```

. . .
. . .

The following table describes some of the actions that can appear.

Action	Description	User Action
activate	Indicates that the user activated an integration.	Activate the integration.
activate_phase_change	Indicates that the user activated an integration. This action captures the intermediate state in the activation.	Activate the integration.
audit_migration	Indicates that the user cloned an instance.	Clone the instance from the Settings page, under Import and Export.
bulk_reactivation_completed	Indicates that the user performed a Save and Reactivate operation from the Connections page.	Save and reactivate the connection.
bulk_reactivation_started	Indicates that the user performed a Save and Reactivate operation from the Connections page.	Save and reactivate the connection.
clone	Indicates that the user cloned a resource (for example, integration, connection, or lookup).	Clone the resource.
configure_integration	Indicates that the user configured an integration from the Configuration Editor page.	Configure the integration.
create	Indicates that the user created a resource.	Create the resource.
deactivate	Indicates that the user deactivated an integration.	Deactivate the integration.
delete	Indicates that the user deleted a resource.	Delete the resource.
delete_certificate	Indicates that the user deleted the certificate.	Delete the certificate.
disable	Indicates that the user disabled a File Server user.	Disable a user.
import	Indicates that the user imported an integration or package.	Import the integration or package.
instance_resubmission	Indicates that the user submitted a request to resubmit an errored instance for an integration.	Resubmit.
instances_resubmission	Indicates that the user submitted a request to resubmit an errored instance for an integration.	Perform a bulk resubmit.
instances_abort	Indicates that the user submitted a request to abort an errored instance for an integration.	Abort.

Action	Description	User Action
instance_abort	Indicates that the user submitted a request to abort an errored instance for an integration.	Perform a bulk abort.
instance_abort	Indicates that the user submitted a request to abort a running instance for an integration.	Abort.
lock	Indicates that the user locked a resource.	Edit the resource.
paused	Indicates that the user paused a schedule for an integration.	Pause the schedule.
reactivation	Not being used.	--
reactivation_fail	Not being used.	--
register	Indicates that the user created an application type (that is, an adapter).	Create the adapter.
rename	Indicates that the user renamed a resource.	Rename a resource.
resumed	Indicates that the user resumed a schedule for an integration.	Resume a schedule.
start	Indicates that the user started File Server.	Start File Server.
started	Indicates that the user started a schedule for an integration.	Start a schedule.
state_change	Indicates that the user activated an integration. This action captures the intermediate state in the activation.	Activate the integration.
stop	Indicates that the user stopped File Server.	Stop File Server.
stop_schedule_deactivation	Not being used.	--
stopped	Indicates that the user stopped a schedule for an integration.	Stop the schedule.
unlock	Indicates that the user unlocked a resource.	Edit and save a resource.
update	Indicates that the user updated a resource.	Update the resource.
update_schedule_parameter	Indicates that the user updated a schedule parameter's value for an integration.	Update the schedule parameter.
upload_certificate	Indicates that the user uploaded the certificate.	Upload / create the certificate.

See the Current Status of All Integration Components

Get at-a-glance information about the number and status of your projects, integrations, connections, and more. For example, learn how many REST connections your organization has created.

This information can be helpful for showing your team's contributions, such as when you're compiling year-end metrics for the work that you've delivered.

- In the navigation pane, click **Observability**, then **Dashboards**.

- Along the top, select the **Design time and audit** box.

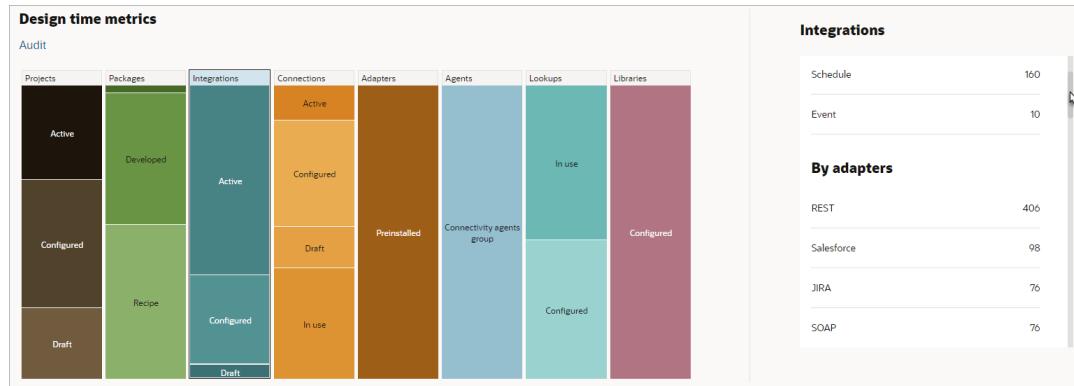
The Design time metrics heading appears. A chart visualizes the components in your Oracle Integration instance.

- Obtain metrics using one or both of the following options:

- Below the **Design time metrics** heading, point to an entry in the chart, and view a metric in the tooltip that appears.
- On the right, below the **Integrations** heading, obtain additional metrics.

 **Tip:**

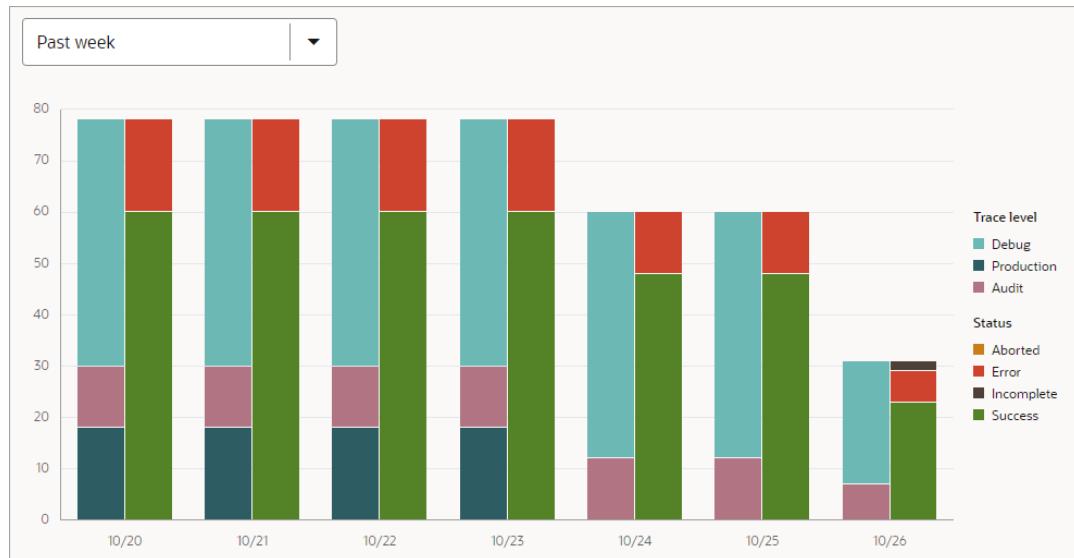
The table of metrics on the right often contains more information than can fit on the page. Point within the table of metrics to see a scroll bar, and scroll down to see additional metrics.



Check an Integration's Tracing Level

To ensure that your integrations are capturing the minimum required tracing information, consider routinely checking their tracing levels. For instance, Oracle recommends not using the Debug tracing level in production.

- In the navigation pane, click **Observability**, then **Dashboards**.
 - Below the **Instance status** heading, select the appropriate time period to review from the drop-down.
- For instance, you might want to check the tracing levels for the last 24 hours or week.
- Familiarize yourself with the chart that appears below **Instance status**.



- Each time period contains two vertical bars.
 - The **left bar** visualizes the **tracing levels** for the integrations that ran during the time period.
 - The **right bar** visualizes the **statuses** for the integrations that ran during the time period.
4. Check the tracing levels for your integration instances for the time period.

View Notification Alert Announcements and the Status of Asynchronous Operations

You can view notification alert announcements and the status of asynchronous operations in the banner.

Topics:

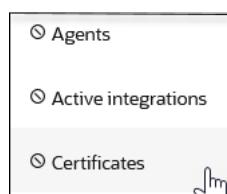
- [View Notification Alert Announcements in the Banner](#)
- [View the Progress of Asynchronous, Design-Time Operations](#)

View Notification Alert Announcements in the Banner

A notification alert announcement icon is displayed in the banner that provides links to the status of connectivity agents, active integrations, and certificate expiration dates.

When an alert is raised, a number is displayed in the alert icon in the banner.

1. Click **Announcements**  in the banner, then select an option to view status. For this example, the following alerts have been raised.



If you select **Certificates**, the Certificate page opens with a filter set to show the soonest-to-expire certificates. In this case, two have expired.

Name	Type	Category	Status	Expiration
expiring6	X.509	Identity	Expired	July 6, 2023
expiring8	X.509	Identity	Expired	July 8, 2023
identity	X.509	Trust	Configured	In 27 Days
twitter_cert	X.509	Trust	Configured	In 1 Months
ygm05ot003_ygn05_mesh-client_nhs_uk-private-key	X.509	Identity	Configured	In 2 Years
YGM05OT003_YGM05_mesh_client_nhs_uk	X.509	Trust	Configured	In 2 Years
NHS_INT_Level_IC	X.509	Trust	Configured	In 6 Years
digi_sha_2020.ca1	X.509	Trust	Configured	In 7 Years

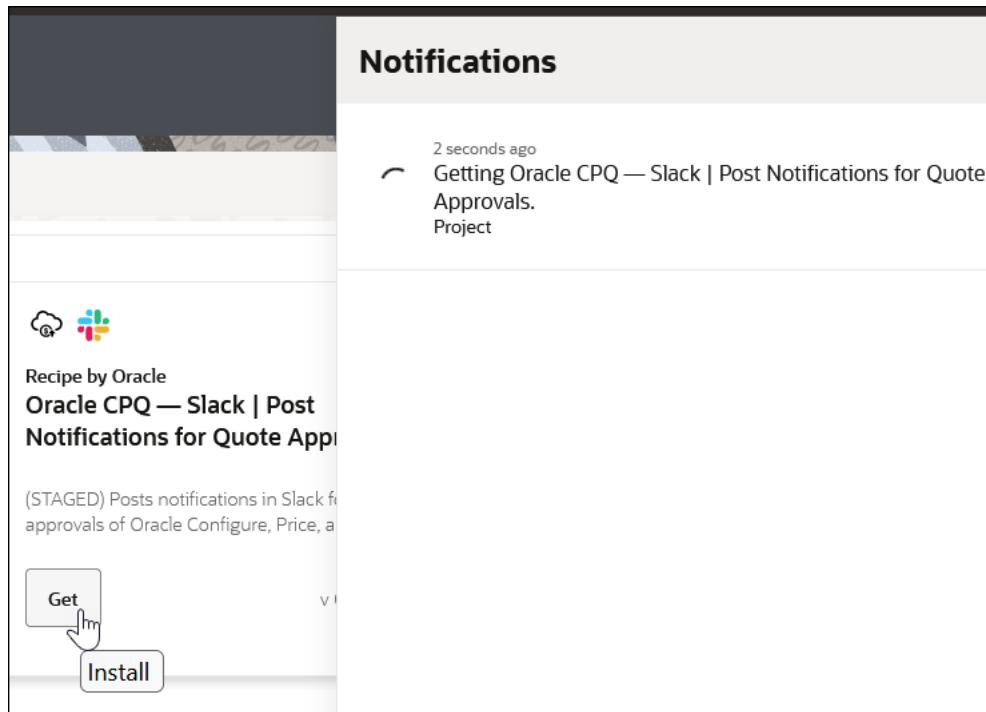
2. If you select **Agents** or **Active integrations**, the following pages are displayed:
 - **Agents**: Takes you to the **Unavailable agents** card of the Integration dashboards page. The percentage of agents running and not running is displayed.
 - **Active integrations**: Takes you to the **Active integrations** card of the Integration dashboards page. The number of active integrations and percentage of active scheduled, event, and application integrations are displayed.
3. Take appropriate action based on the information provided.

View the Progress of Asynchronous, Design-Time Operations

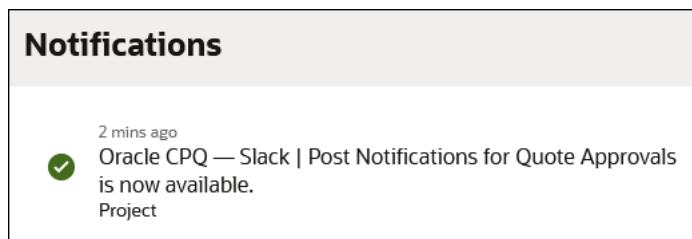
You can view the current progress of some asynchronous, design-time operations from the notifications center icon in the Oracle Integration banner. This view eliminates the need to constantly refresh the page to check progress.

You can view the progress for the following asynchronous operations from the notifications center:

- Install accelerator projects
 - Import projects
 - Deploy integrations to the Oracle Cloud Infrastructure API Gateway
1. Perform one of the supported asynchronous operations.
 2. Click **Notifications**  in the Oracle Integration banner.
 3. Follow the real-time progress. For this example, a recipe is being installed.



- Reopen the notification center. The progress message has changed to indicate that installation has completed. The message is retained for six hours.



- Click the recipe name in the notification center to access the project.

Monitor the Message Processing Status of Integrations

The Monitor integrations page enables you to view the message processing status of your running integrations, including how many messages have been received and processed, how many successful messages and errors have occurred, and how many messages have been aborted.

Only activated integrations are listed on this page by default. Using the appropriate filter, you can also view data for inactive integrations.

- In the navigation pane, click **Observability**, then **Integrations**.

A list of active integrations appears by default, along with processing information about the number of messages received, the number of messages processed, the number of successful messages, the number of failed messages, and the number of aborted messages. For schedule integrations, navigation links are provided that take you to the Schedule and Future Runs page.

A **Total instance count** summary is displayed at the top of the page. This section is collapsed by default. Data is not immediately fetched.

2. Expand the **Total instance count** section and click the **Refresh** button to fetch and view data.

This summary shows the count of total available messages in various states present in the tracking store. This summary is different from the metrics shown on the Integration dashboards page. While the Integration dashboards page metrics consist of historical data collected since the environment was created, the Monitor Integrations page shows only the current set of messages that have not yet been purged.

Received	Processed	Succeeded	Errorred	Aborted
11890	11856	5345	6256	255

3. Click **Filter**, then select the time window, integration status (active, configured, activating, failed, and all), integration style (application, schedule, or event), or integration type (all integrations, only integrations in a project, or only integrations not in a project). If you select to show integrations in a project, you are prompted to begin entering the project name and select from the matching suggestions.

You can also specify a custom time range to filter the search for integrations under **Date range** with the **From** and **To** buttons. This functionality also exists on other monitoring pages such as the Instances page or Errors page.

Name	Received	Processed	Succeeded	Errorred	Aborted	Last scheduled by
testWait (1.0) Running Last message: Just now	4609	4615	4605	0	10	l2serviceadmin
Zendesk_3Leg_Ticket_Create_AND_Inbound_G3 (1.0) Last message: 2 minutes ago	40	40	40	0	0	
zendesk_inbound0805 (1.0) Last message: 2 minutes ago	31	31	31	0	0	
MON_SCHEDULER_TRACKING (1.0) Running Project: MON_PROJECT_TESTING Last message: 7 minutes ago	768	768	768	0	0	l2serviceadmin
Shopify_Create_Customer_Inbound (1.0) Project: SHOPIFY_AUTOMATION Last message: 35 minutes ago	348	348	348	0	0	
SSP_Shopify_2022-10_Auto1 (1.0) Project: PI_SHOPIFY Last message: 39 minutes ago	245	245	245	0	0	

The values shown in the table columns are based on the selected time window (last 1 hour, last 6 hours, and so on):

- **Received:** Shows the number of new incoming messages (or instances created). It is possible that some of these messages will not complete processing within the selected time window.
- **Processed:** Shows the sum of the **Succeeded**, **Errorred**, and **Aborted** column values. This is the number of messages that completed processing in the selected time

window. Some messages may have been received in a prior time window, but completed processing during the currently-selected window.

- **Succeeded, Errorred, and Aborted:** The explanation is the same as the **Processed** column count. The only difference is the different terminal states.

A difference can exist between the number of messages listed in the **Received** and **Processed** columns. For example, assume there are 76 **Received** column messages and 184 **Processed** column messages. This means that for the selected time window, 76 new messages were received. In that same time window, 184 messages completed processing. Many of the 184 would have been received before the selected time window, and therefore not counted in the **Received** message total. Similarly, many of the 76 **Received** messages may not complete within the selected time window. Therefore, even though they are counted in the **Received** total, they are not counted in the **Processed** total.

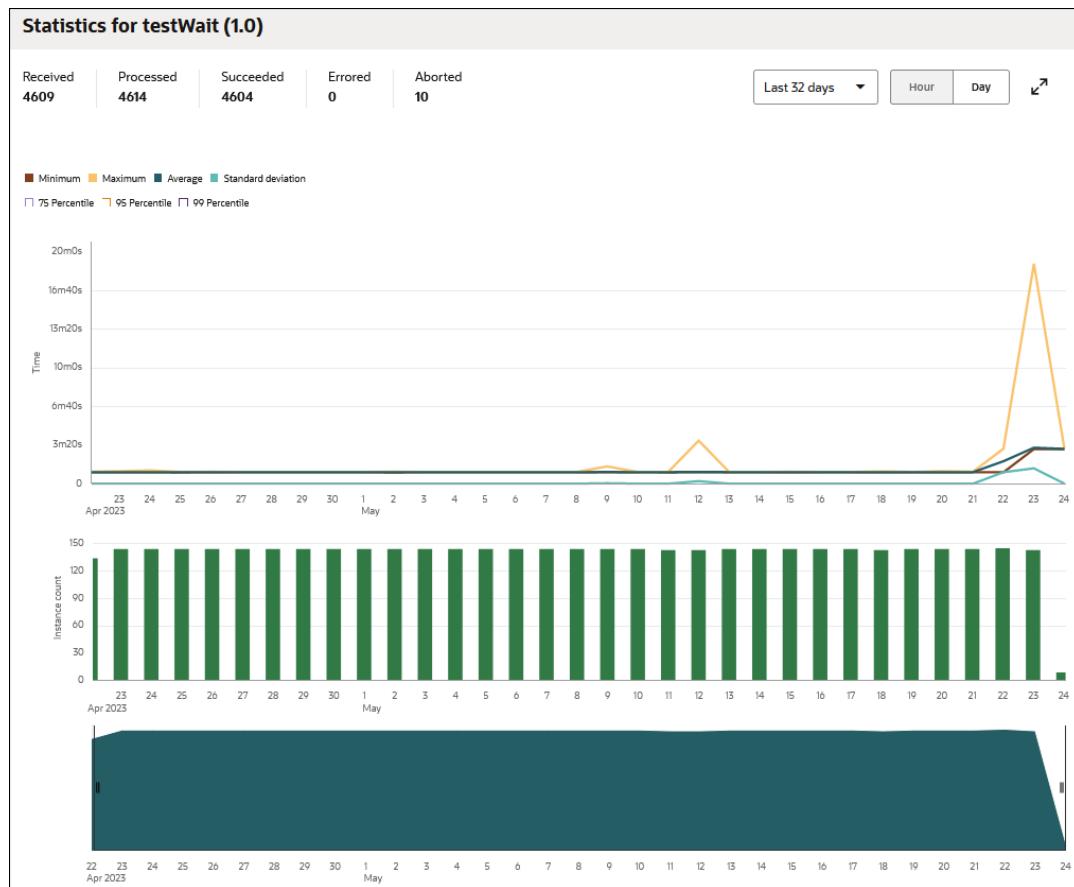


For schedule integrations (identified by ) , the **Name** column includes a label identifying the state of the schedule integration:

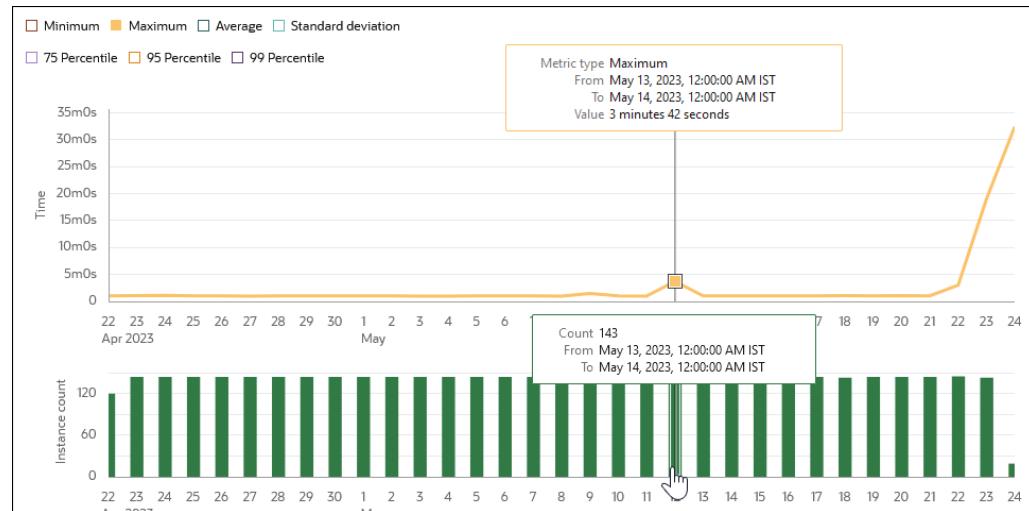
- No schedule defined
- Paused
- Running
- Started
- Stopped

The user who last submitted the schedule of an integration is displayed in the **Last scheduled by** column. For nonschedule integrations such as application integrations, this column remains empty.

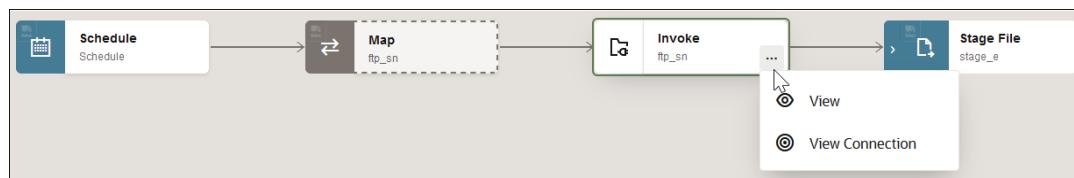
4. Click numbers in the **Received**, **Processed**, **Succeeded**, and **Aborted** columns to access the list of instances in those states on the Instances page. If you click the **Received** number, instances are displayed based on their creation date. All other clickable numbers are sorted by their last updated date.
5. Click numbers in the **Errorred** column to access the list of instances in that state on the Errors page.
6. Hover over a row for an integration and click **View statistics**  to show aggregated instance metrics for the selected integration. You can view the total counts and the minimum (quickest), maximum (slowest), average (mean), and standard (largest) deviation execution times of integrations over a selected time period (for example, last hour, last 32 hours, last week, and more). You can also select to view which of the execution times fall into the 75th, 95th, and 99th percentiles.



- a. Select the check boxes for which to view execution time metrics. By default, each check box is selected.
 - **Minimum:** Minimum (quickest) integration execution times.
 - **Maximum:** Maximum (slowest) integration execution times.
 - **Average:** Average (mean) integration execution times.
 - **Standard deviation:** Standard (largest) deviation execution times.
 - b. If you want to know the percentile into which the selected metric falls, select a percentile.
 - c. Select a time range for which to view metrics.
 - d. Drag your cursor across the time segments in the **Instance count** section or drag the vertical bar across the **Time** section to show instance execution metrics for a particular time period. These metrics can help identify execution trends. For example, execution times are slower at one time, which may occur because of an adapter connection issue. Or there is a significant wait time for schedule integrations to run because too many schedule integrations are executing at once. You can then revise the schedule to make more optimal use of Oracle Integration.



- e. Click a time segment bar in the **Integration count** section to access a filtered list of the instances for that segment on the Instances page.
- 7. Click an integration in the **Name** column to access a graphical view of the integration flow. The **Share** drop-down list provides options for exporting the integration or printing a graphical view of the integration. The inline menus of the triggers, invokes, and actions in the integration flow provide access to additional details. For example, for an invoke connection:



- 8. Click **Actions ...** to view a set of options. The options that appear are based on the trigger, action, or invoke that you selected.
 - a. For actions, click **View** to view configuration details.
 - b. For triggers and invokes, click **View** to view Adapter Endpoint Configuration Wizard configuration details and **View connection** to view connection configuration details.
 - c. For schedules, click **View** to view configured schedule parameters or **Edit schedule definition** to edit the schedule definition.
 - a. For logic actions such as a scope, switch, for each, and others, click **Expand** to view all elements within the action or **Collapse** to close the logic action.
 - b. For scopes, click **Fault handlers** to view any fault handlers defined in the scope.
- 9. For schedule integrations (identified by a calendar icon), hover over the row of the integration to access and click **Actions ...**. The tasks available for selection are based upon the state of the schedule integration.

Schedule Integration State	Actions Menu Items
No schedule defined	<ul style="list-style-type: none">• Add schedule• Run
Paused	<ul style="list-style-type: none">• Run• Stop schedule• Resume schedule• Schedule
Running	<ul style="list-style-type: none">• Pause schedule• Run• Schedule• Stop schedule• View future runs
Started	<ul style="list-style-type: none">• Retry• Run
Stopped	<ul style="list-style-type: none">• Edit schedule• Run• Schedule• Start schedule

 **Note:**

If you are a user with the ServiceAdministrator role, you can change the user with which to run the schedule.

10. For schedule integrations, hover over a row and select **Open Details**  to view details about the time of the last message and the time for the next scheduled run.

Monitor Agents

When the Dashboard tells you that one or more connectivity agents are unavailable, come to the Agents page. Here, you can determine which connectivity agents and agent groups are down.

For details about monitoring the environment that robot instances run on, see Check Whether an Environment is Available in *Using Robots in Oracle Integration 3*.

1. In the navigation pane, click **Observability**, then **Agents**.

Name	Type	Status	Agents
▶ Finance agents	Connectivity agents...	Partially Available	2
▶ Human resources agents	Connectivity agents...	Unavailable	1
Development agents	Connectivity agents...	Unavailable	0
▶ Executive agents	Connectivity agents...	Unavailable	1
▶ Test agents	Connectivity agents...	Unavailable	1

The Monitor Agents page shows details such as the time at which the agents were last updated and the connectivity agents associated with the agent groups.

- If the agent status is green, this indicates that the agent is running and able to process messages.
 - If the agent status is red, you cannot select it in the Select an Agent Group dialog of the Connections page, even though the agent may be up and running.
2. Click **Expand ▾** to view details about the agent group such as the creation date and the last startup date for the connectivity agent.

Note:

When you shut down the connectivity agent server, several minutes pass before the agent is displayed as down on the Monitor Agents page (as indicated by a red thunderbolt).

Learn About the Payloads in the Activity Stream

The activity stream shows details about how messages move through an integration, including whether failures occurred. The activity stream in the user interface shows only messages of certain sizes and types. However, you can see all payloads at any time by downloading the activity stream.

- [Types of Payload Messages That Appear in the Activity Stream](#)
- [Payload Messages That Appear in the Activity Stream](#)

Types of Payload Messages That Appear in the Activity Stream

The activity stream provides an inline view of some payloads. However, binary payloads, such as a zip file, cannot be viewed in the activity stream. Instead, you can directly download those payloads for viewing.

Activity Stream

Instance Id: GLEnMcn7EeyxgOnHEzvB-A

Trace with payload

Mon May 02 06:34:42.988 PM KST 2022 Wire Message received by Trigger EmployeeAPI 32ms

06:34:43.020 PM Message started processing 1ms

`<execute
xmlns="http://xmlns.oracle.com/cloud/adapter/REST/EmployeeAPI_REQ
UEST/getEmployee/types">
 <HTTPHeaders />
</execute>`

Payloads with the following media types (MIME types) can be viewed in the activity stream:

- text/javascript
- text/typescript
- text/strings
- text/html
- text/x-yaml
- text/css
- text/x-sql
- text/x-mysql
- text/x-sh
- text/plain
- text/xml
- application/json
- application/xml
- application/typescript
- application/javascript

For all other media types, download the payloads directly to your system by clicking **Download**

at the top of the activity stream to download the entire activity stream of the instance or inline to download the payload for the selected section of the activity stream.

The screenshot shows the Activity Stream interface with the title "Activity Stream" and an instance ID. It displays a trace with payload, showing a sequence of events:

- Wire Message received by Trigger echoRequest (Mon May 02 06:46:04.172 PM KST 2022) - 2s 326ms
- Message started processing (06:46:06.498 PM) - 373ms
- Note: The payload is in a format which cannot be displayed. Please click the Download Payload button to download.
- Message received by Assignment assignAction (06:46:06.871 PM)

Payload Messages That Appear in the Activity Stream

The size of the payload determines whether it can be viewed in the activity stream or must be downloaded for viewing.

Attachments are not considered part of the payload and are not visible in the activity stream. Adapters store attachments and delete them after processing completes.

Payload Sizes	Viewable in the Activity Stream
Small: Less than 32 KB	Yes (visible inline)
Large: More than 32 KB	No (can be downloaded from the activity stream)

See [Track the Status of Integration Instances](#) and [View Error Messages and the Message Payload in the Activity Stream of Failed Instances](#).

Measure an Integration's Performance

Expect to measure an integration's performance often, including while designing the integration, before promoting the integration to production, and periodically after the integration is in production.

Perform any or all of the following tasks to measure an integration's performance. If you're troubleshooting, these steps help you determine the activities that are taking the most time and analyze how many integration instances have run for longer than expected. This information can help you determine your next steps.

What Do You Want to Do?

- [Determine an Integration's Typical Processing Times](#)
- [Determine an Integration's Performance on a Given Day](#)
- [Determine Whether an Integration's Performance Is Consistent Over Time](#)

- Find Integration Instances with Increased Processing Times
- Measure the Running Times of an Integration's Actions

Determine an Integration's Typical Processing Times

You should know your integrations' typical running times so that you can more easily identify the integration instances that took longer than expected to run.

You have two options for checking processing times: review numerical times on the Instances page, or review graphical information on the Integrations page.

How Oracle Integration Calculates Duration

Oracle Integration calculates an integration instance's duration in the following ways:

- Schedule integrations: Time that passes from start to finish.
- Failed integrations: Time for all activities, including sending requests and rerunning the integration.
- All other integrations: Time that passes between the triggering event and the flow finishing.

Option 1: Review Graphical Information

1. In the navigation pane, click **Observability**, then **Integrations**.
2. Hover over the row of an integration, and select **View processing time statistics** .

The Processing time panel opens. The panel shows the processing time for all integration instances for the integration.

3. If needed, update the time filters at the top of the panel.

Received 173	Processed 173	Succeeded 173	Errored 0	Aborted 0	Last 32 days 	By hour	By day
------------------------	-------------------------	-------------------------	---------------------	---------------------	----------------------------------------------------------------------------------------------------	---------	--------

4. Above the chart, select the following values to hide the lines for the values:
 - **Minimum**
 - **Maximum**
 - **Standard deviation**

Only the Average value remains.

5. Review the average processing times for the integration over time.

Option 2: Review Numerical Times

1. In the navigation pane, click **Observability**, then **Instances**.

2. Click **Filter** 

For example, from the **Sort by** drop-down, select **Duration**. You might also want to update the time window filter.

3. For each integration instance, find the **Duration** column, and review the value in the **Duration** field.

Determine an Integration's Performance on a Given Day

You can quickly measure an integration's performance for a given day.

1. In the navigation pane, click **Observability**, then **Integrations**.

2. Hover over the row of an integration, and select **View processing time statistics** .

The Processing time panel opens. The panel shows the processing time for all integration instances for the integration.

3. If needed, scroll down until you see the Instances chart, which is the second graphical chart on the page.

4. Point to the vertical line for a given date, such as a date with a spike in its processing times, and click the line.

The Instances page appears and shows only the integration instances that ran on the selected date. The integration instances are sorted by instance duration, with the longest durations appearing first.

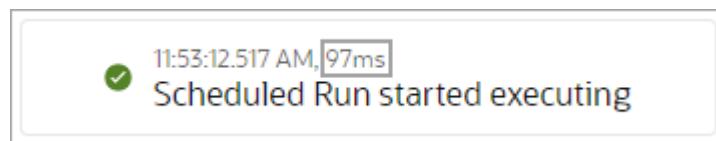
5. Optional: Identify the activity that required a longer-than-usual processing time.

- a. For an instance with a longer-than-expected duration, hover over the row, and select **View details** .

The activity stream opens.

- b. Review the processing time for each part of the integration flow to determine which steps took the most time.

The processing time appears after the timestamp, above the information about the activity.



- c. Identify the activity that required a longer than usual processing time.

Determine Whether an Integration's Performance Is Consistent Over Time

You typically verify that an integration meets your organization's performance requirements before promoting the integration to production and then periodically after the integration is in production.

1. In the navigation pane, click **Observability**, then **Integrations**.

2. Hover over the row of an integration, and select **View processing time statistics** .

The Processing time panel opens. The panel shows the processing time for all integration instances for the integration.

3. If needed, update the time filters at the top of the panel.



4. Check whether the standard deviation varies from the average performance time.

Ideally, the values for these lines should be similar. If they're not, a number of integration instances might take longer than expected to run.

- a. Above the first chart, select the following values to hide the lines for the values:

- **Minimum**
- **Maximum**

Only the **Average** and **Standard deviation** values remain.

- b. Compare the **Average** and **Standard deviation** lines.

Similar values indicate that the integration instances have consistent durations. When the lines aren't similar, consider investigating the outliers.

5. Check for spikes in the the maximum and average performance times.

Ideally, the values for the **Maximum** and **Standard deviation** lines should remain low over time. When these values spike, some integrations are taking longer than usual to run.

- a. Hide the **Average** line, and show the **Maximum** line.

Only the Maximum and Standard deviation values remain.

- b. Review the **Maximum** and **Standard deviation** lines.

Low values indicate that integration instances for this integration have similar processing times. On the other hand, spikes in the lines indicate that some instances have taken longer than usual to process. Consider investigating these spikes.

Find Integration Instances with Increased Processing Times

Charts and graphs help you quickly determine whether a slow-performing integration instance is an outlier or a trend.

1. In the navigation pane, click **Observability**, then **Integrations**.

2. Hover over the row of an integration, and select **View processing time statistics** .

The Processing time panel opens. The panel shows the processing time for all integration instances for the integration.

3. Above the first chart, select the following values to hide the lines for the values:

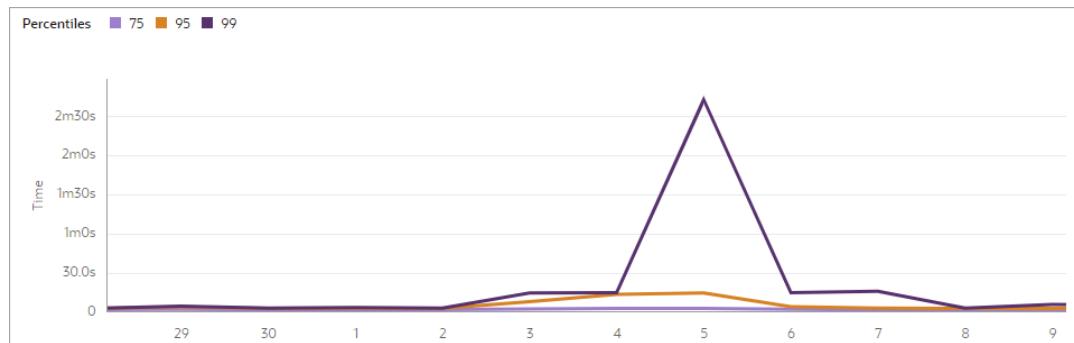
- **Minimum**
- **Maximum**
- **Average**
- **Standard deviation**

4. Above the first chart, select the following values to show the lines for the values:

- **75**
- **95**
- **99**

5. Point to the spikes for the each of the percentile lines, and view their processing times in a popup.

These numbers offer insights into the processing times of integrations. For example, consider the following chart.



- The 75 line indicates that 75 percent of instances of this integration finished in about 3 seconds or less.
- The 95 line indicates that 95 percent of instances of this integration finished in about 15 seconds or less.
- The 99 line indicates that 99 percent of instances of this integration finished in about 2 minutes, 30 seconds or less.

- Determine whether the performance for the integration has remained consistent.

Measure the Running Times of an Integration's Actions

When an integration instance takes longer than expected to finish, determining the actions that took the most time can help you troubleshoot.

- In the navigation pane, click **Observability**, then **Instances**.
 - Click **Filter** , and update the filtering requirements for the page.
 - Hover over the row of an integration instance, and select **View details** .
- The activity stream opens.
- Review the duration for each part of the integration flow to determine which steps took the most time.

The processing time appears after the timestamp, above the information about the activity.



Find Integrations That Experienced Errors

When you're responsible for monitoring integrations, part of your job involves determining whether any integrations experienced errors and then taking the appropriate next steps.

Errors in live integrations are never expected. Oracle recommends investigating every error that an integration experiences.

What Do You Want to Do?

- Find Integrations with Errors
- Determine the Source of an Integration's Error
- Find the Integration Error Rate for the Last 24 Hours
- Measure Your Integration Errors Over Time
- Find the Connections with the Most Errors

Find Integrations with Errors

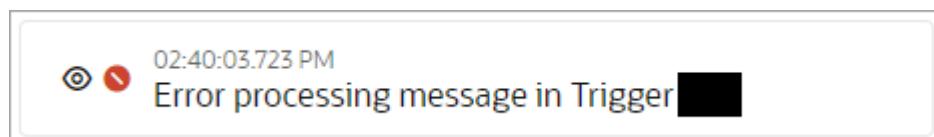
Oracle recommends investigating every error that occurs with your integrations.

1. In the navigation pane, click **Observability**, then **Integrations**.
2. Click **Filter** , and update the filtering requirements for the page.
3. In the **Errored** column, look for any non-zero values.

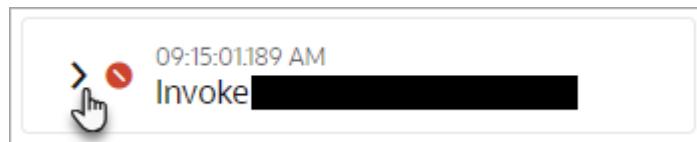
Determine the Source of an Integration's Error

Find the source of an error by checking the activity stream. You can also see the location of the error within an integration flow on the integration canvas.

1. In the navigation pane, click **Observability**, then **Integrations**.
2. Click **Filter** , and update the filtering requirements for the page.
3. In the **Errored** column, look for any non-zero values.
4. For an integration with one or more errors, determine the location of the error(s).
 - a. Click the number value in the **Errored** column for the integration.
The Errors page appears and shows only the integration instances that experienced errors.
 - b. For an error, review the value in the **Fault location** column.
The fault location indicates where the error occurred. For example, if errors occurred for an invoke that multiple integration instances use, the application that the invoke connection is connecting to might be unavailable.
5. Locate the source of the issue by reviewing the activity stream.
 - a. Hover over an errored integration instance, and select **View details** .
 - b. In the activity stream, find the milestone with an error and open the activity stream from it using one of the following options.
 - If **View payload**  appears on the milestone, click **View payload** .



- If **View payload** ⓘ doesn't appear on the milestone, expand the milestone by clicking the expand icon >, and then click **View payload** ⓘ for the part of the integration flow that failed.

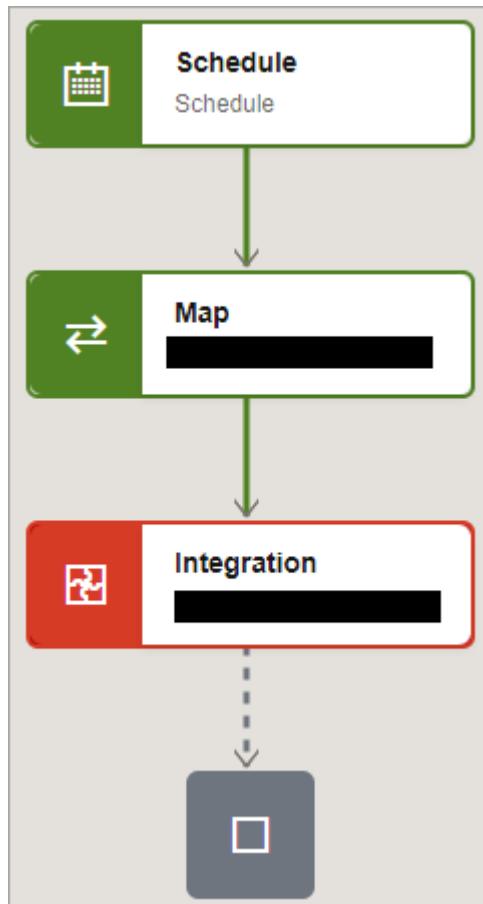


- c. Review the payload and take the appropriate action.

For example, if a 404 error occurred, check the configuration of the connection.

6. Optional: To see the error within the integration flow, open the integration canvas.
 - a. On the Errors page, in the **Primary identifier** column, click the name of the primary identifier, such as **Primary: undefined**.

The integration canvas appears. The integration component with an error is marked in red.



- b. To see the error in the activity stream: Point to the component with an error, click **Actions ...**, and select **Activity stream**.

The activity stream opens.

7. If you are not responsible for addressing the error, send the error information to the owner of the integration.

For example, you might perform some or all of the following steps:

- Take a screenshot of the activity stream.
- Copy the payload text.
- Download the logs for the activity stream.

 **Tip:**

Click **Download Logs**  at the top of the activity stream.

Find the Integration Error Rate for the Last 24 Hours

Consider completing a daily check of your integration error rate over the last 24 hours. If the value is greater than 0, investigate the errors.

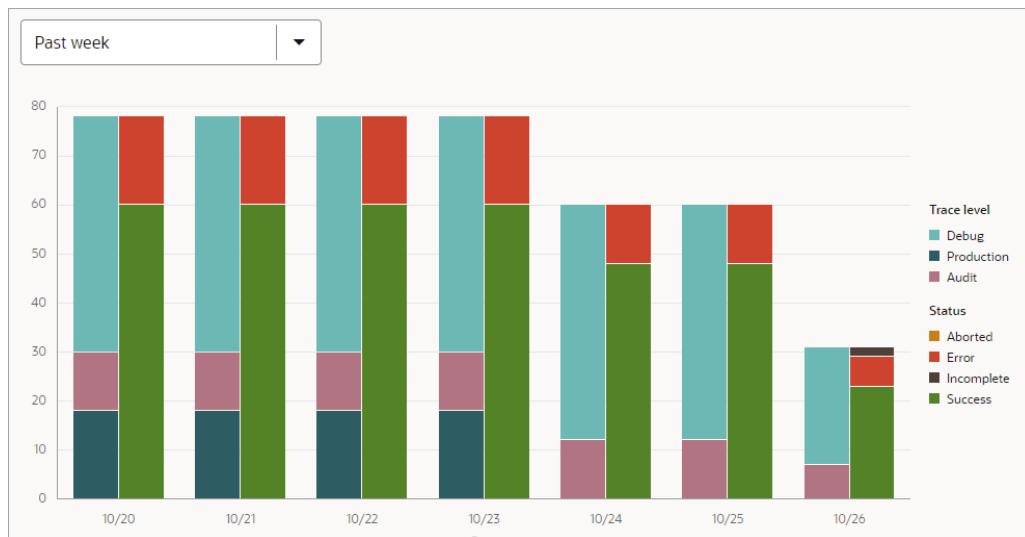
1. In the navigation pane, click **Observability**, then **Dashboards**.
2. Along the top of the page, make sure that **Error rate (past 24 hours)** is selected.
3. Review the value in the **Error rate (past 24 hours)** box.

This metric shows the percentage of integration instances that experienced errors in the last 24 hours.

Measure Your Integration Errors Over Time

Measuring your integration errors over time can help you track trends and measuring improvements.

1. In the navigation pane, click **Observability**, then **Dashboards**.
2. Below the **Instance status** header, use the drop-down to adjust the time period, if necessary.
3. Review a chart to visually track your errored integrations over time.
 - a. Familiarize yourself with the chart that appears below **Instance status**.



- Each time period contains two vertical bars.
 - The **left bar** visualizes the **tracing levels** for the integrations that ran during the time period.
 - The **right bar** visualizes the **statuses** for the integrations that ran during the time period.
- b. Determine whether any integration instances experienced errors during the time period.
4. Review the metrics to understand how many integration instances have experienced errors over time.

The error-related metrics appear to the right of the chart, below the **Overview** heading.

- **Error rate:** Error rate for all of the integration instances during the time period.
- **Instances:** Number of integration instances that ran during the time period.
- **Errors:** Number of integration instances with a state of Errored during the time period.

Find the Connections with the Most Errors

Knowing the connections that result in integration instances with errors can help you quickly prioritize integration failures and start the troubleshooting process.

Why Connections Experience Errors

The typical causes of connection errors are varied and depend upon the adapters you use, the way you use the connections, and your use cases. For example:

- The integration developer configured the connection incorrectly.
In this case, the connection always fails.
- The connection cannot complete its task.

Consider a connection that is based on the FTP Adapter. The connection fails if someone moves or deletes the folder that the integration reads from or changes the permissions on the folder.

To Find the Connections with the Most Errors:

1. In the navigation pane, click **Observability**, then **Dashboards**.
2. Along the top, select the **Connections with errors** metric.

The **Top connections with errors** section appears, listing up to five of the connections with the most errors.

3. Below the **Top connections with errors** heading, point to the box for a connection.

A tooltip displays the number of integration instances that use the connection and experienced an error.

4. Find the integration instances that used the connection and experienced errors.

- a. Click within one of the connections boxes.

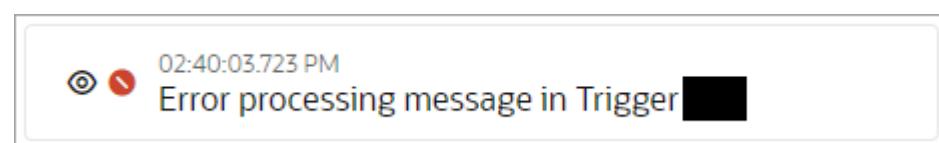
The Errors page appears and shows only the integration instances that used the connection and experienced an error.

- b. Hover over an errored integration instance, and select **View details** .

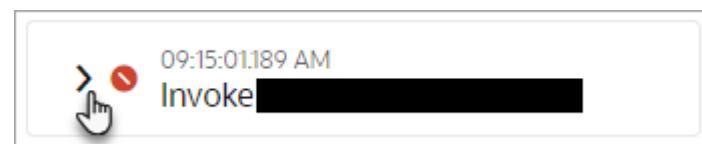
The activity stream opens.

- c. In the activity stream, find the milestone with an error and open the activity stream from it using one of the following options.

- If **View payload**  appears on the milestone, click **View payload** .



- If **View payload**  doesn't appear on the milestone, expand the milestone by clicking the expand icon >, and then click **View payload**  for the part of the integration flow that failed.



- d. Review the payload and take the appropriate action.

For example, if a 404 error occurred, check the configuration of the connection.

Find Integrations That Timed Out or Were Aborted

Being able to quickly find an integration that has timed out or been aborted can help you troubleshoot.

Why Integration Instances Have a Status of Aborted

Several activities can change an integration instance's status to Aborted, including the following:

- Someone canceled the integration.
- Someone stopped the scheduled of a scheduled integration.
- Someone edited a connection for a queued or in progress schedule integration.
- The integration timed out.

Time Limits for Integrations

Different integrations can run for different amounts of time. To see the time limits, see Service Limits in *Provisioning and Administering Oracle Integration 3*.

To Find All Aborted Integrations:

You can work on either the Instances or Integrations page.

- Find all aborted integrations using one of the following options.
 - Filter the Instances page so it shows only aborted integrations.
 1. In the navigation pane, click **Observability**, then **Instances**.
 2. Click **Filter** , and from the **Status** dropdown, select **Aborted**.
 3. Update the other filters as needed, and click **Apply**.The page shows only aborted integrations.
 - Scan the Integrations page and look for any integrations with one or more aborted instances.
 1. In the navigation pane, click **Observability**, then **Integrations**.
 2. Review the **Aborted** column for any non-zero values.An integration with a non-zero value in the **Aborted** column has one or more aborted integration instances.

Next, determine why an integration was aborted. See [Determine Why an Integration Instance Stopped Running](#).

Manage Errors

You can manage errors from the Errors pages in Oracle Integration. You can resubmit failed instances, discard failed instances, view message recovery status, and view basic and detailed error messages. You can also view and download the activity stream, which provides access to error messages and message payloads in failed instances.

Topics:

- [View Error Messages and the Message Payload in the Activity Stream of Failed Instances](#)

- Resubmit Failed Messages
- Discard Errors
- View the Status of Message Recovery
- View Errors By Integration and Connection
- View the Business Identifiers Used in Failed Integration Instances

See [About Error Management](#).

View Error Messages and the Message Payload in the Activity Stream of Failed Instances

You can view error messages and the message payload in the activity stream of failed integration instances on the Errors page. You can also download the activity stream logs and the payload of the failed section of an integration instance.

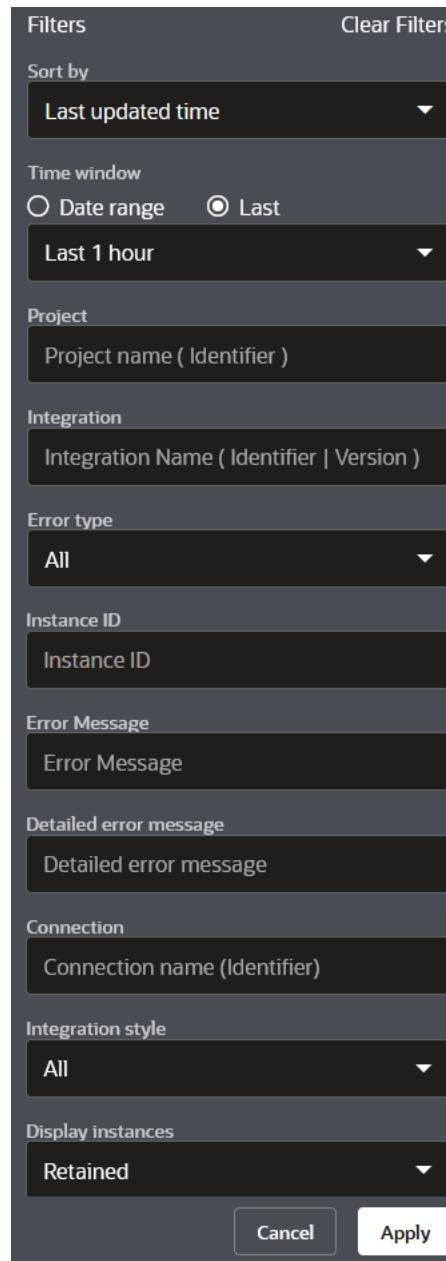
1. In the navigation pane, click **Observability**, then **Errors**.
2. Perform the following tasks:
 - Filter the Display of Error Message Details
 - View Error Message and Payload Details on the Errors Page
 - View the Flow of the Failed Instance and Payload Errors in the Activity Stream on the Instance Details Page

Filter the Display of Error Message Details

1. Click **Filter**  to filter the display of errors by the following:
 - Last updated or creation time
 - Project (begin entering a name to display a list from which to select)
 - Integration
Begin typing an integration name. Do not use copy and paste. The filter is refreshed to display fields for the defined secondary business identifiers.

You can now begin typing the values of the business identifiers to narrow down the search. This input follows the same rules (for example, for exact matches and phrase matches) as described in [Search for Primary and Secondary Business Identifier Values](#).
 - Error type (all, recoverable, or nonrecoverable)
 - Instance ID
 - Error message
For example:
 - Enter part of an error message, such as unknown, unauthorized, or endpoint.
 - Enter an exact phrase match of an error message in double quotes, such as "Could not activate endpoint for endpointID"
 - Detailed error message
 - Connection name
 - Integration style (all, application, schedule, or event)
 - Integration state (all, retained, or purged)

When you enter an integration name, the filter is refreshed to display fields for the defined business identifiers. You can enter variable values with special characters or variable values of multiple words with white space in double quotes (for example, &filename or "file tracking").



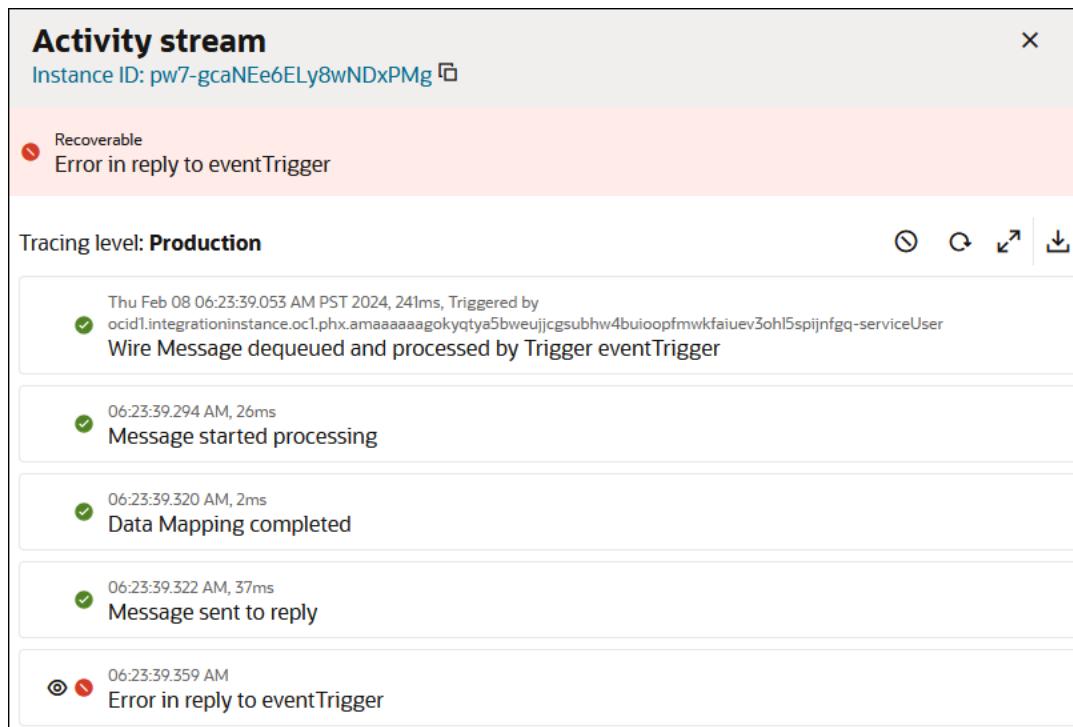
2. Click **Date Range** in the **Time Window** section.
3. Specify a custom time range with which to filter the search for errors with the **From** and **To** buttons. This functionality also exists on other monitoring pages such as the Instances page and Monitor Integrations page.

View Error Message and Payload Details on the Errors Page

Error message details can be displayed in two parts: a summarized error message for easy understanding and resolution and, if necessary, a more specific error message if more detailed troubleshooting is required. You can view error messages on the Errors page or by clicking on

a business identifier to view the entire integration flow on the Instance Details page. You can also view the payload details in the activity stream of a failed instance.

1. Hover over the row of the failed instance.
2. Click **View** . The Activity Stream panel opens. The activity stream shows details about the movement of the message through the integration, including where any failures occurred. The date and time according to your user preferences are displayed.



The screenshot shows the 'Activity stream' panel for an instance with ID 'pw7-gcaNEe6ELy8wNDxPMg'. The panel has a pink header bar with the title and instance ID. Below it, a red banner indicates a 'Recoverable Error in reply to eventTrigger'. The main area shows a list of milestones with timestamps and descriptions, some with green checkmarks and others with error icons. At the top right, there are filter and export icons. A large download icon is at the bottom left.

Milestone	Timestamp	Description
Wire Message dequeued and processed by Trigger eventTrigger	Thu Feb 08 06:23:39.053 AM PST 2024, 241ms	Triggered by ocid1.integrationinstance.oc1.phx.amaaaaaagokyqtya5bweujcgsuhw4buioopfmwkfaiuev3ohl5spijnfqq-serviceUser
Message started processing	06:23:39.294 AM, 26ms	
Data Mapping completed	06:23:39.320 AM, 2ms	
Message sent to reply	06:23:39.322 AM, 37ms	
Error in reply to eventTrigger	06:23:39.359 AM	EventTrigger

3. Click **Download Logs**  at the top of the activity stream to download the activity stream logs for the entire instance. The Oracle Cloud identifier (OCID) value of your service instance is also included in the logs.
4. Scroll through, expand milestones, and click **View Payload**  to view the payload for the part of the integration flow that failed.

The screenshot shows the Oracle Activity Stream interface. At the top, it displays "Activity stream" and "Instance Id: Z18f-HAuEe29jqqliZ3l5g". A red error message "Error processing message in Invoke callAsyncChild" is highlighted with the text "Recoverable". Below this, the tracing level is set to "Debug". The activity log shows several events:

- "Message started processing" (green checkmark)
- "Message entered Scope mainScope" (green checkmark)
- "Message received by Assignment localcounter" (green checkmark)
- "While whileLessThanLoop" (red exclamation mark) with "1 Iteration"

A detailed error view is expanded for the last event:

- Timestamp: 08:41:10.804 AM
- Error type: Error processing message in Invoke callAsyncChild
- Details:

```
oracle.cloud.connector.api.RemoteApplicationException: Fault Details :<ns2:APIInvocationError xmlns:ns2="http://xmlns.oracle.com/cloud/generic/rest/fault/collocatedics/callAsyncChild"><ns2:type /><ns2:title /><ns2:detail /><ns2:errorCode /><ns2:errorDetails><ns2:type>http://www.w3.org/Protocols/rfc2616/rfc2616-
```

An option is provided in the activity stream for quickly showing all error content only. This option can be especially useful for instances in which a logic action such as a for-each loop or while loop with many iterations has failed. This option enables you to jump quickly to the iteration that failed instead of scrolling through each iteration of the loop to access the errors. Show errors in either of two ways:

a. Click **View Errors**



to view only the errors in the activity stream. This persists the selection of errors in the activity stream across multiple instances until you deselect the button. This button enables you to toggle between showing only errors and showing all content in the activity stream.

Activity stream

Instance Id: ZI8f-HAuEe29jqvliZ3l5g

Recoverable
Error processing message in Invoke callAsyncChild

Tracing level: Debug

- 08:41:00.774 AM
 - While whileLessThanLoop
 - Iteration: 1
 - 08:41:10.780 AM
 - Invoke callAsyncChild
 - 08:41:10.804 AM
 - Error processing message in Invoke callAsyncChild

or

- b. Click the error in the activity stream banner to jump to that specific error.

Activity stream

Instance Id: ZI8f-HAuEe29jqvliZ3l5g

Recoverable
Error processing message in Invoke callAsyncChild

Tracing level: Debug

Message started processing

08:41:00.773 AM
Message entered Scope mainScope

> 08:41:00.773 AM
Message received by Assignment localcounter

08:41:00.774 AM

- While whileLessThanLoop
 - Iteration: 1
 - 08:41:10.780 AM, 8ms
 - Message received by Invoke callAsyncChild
 - 08:41:10.788 AM, 16ms
 - Wire Message sent to Local Integration callAsyncChild

08:41:10.804 AM

- Error processing message in Invoke callAsyncChild

Summary **Details**

```
oracle.cloud.connector.api.RemoteApplicationException: Fault Details :  
<ns2:APIInvocationError xmlns:ns2="http://xmlns.oracle.com/cloud/generic/rest/fault/  
/collocateddics/callAsyncChild">
```

5. Perform tasks within the failed part of the payload.

Element	Description
Download Payload 	Click to download this part of the payload to your desktop.
Copy Payload 	Click to copy this part of the payload for pasting into a document.
Toggle Line Numbers 	Click to show or hide line numbers in the payload.

or

1. Hover over the row of the failed instance.
2. Click **Expand**  to view a summarized version of the error message for easy understanding and a detailed version that describes how to resolve the error.

```

Error Message
oracle.cloud.connector.api.RemoteApplicationException: Fault Details :
<ns2:APIInvocationError xmlns:ns2="http://xmlns.oracle.com/cloud/generic/rest/fault/REST/getFilename"><ns2:type/><ns2:title/><ns2:detail/><ns2:errorCode/><ns2:errorDetails/><ns2:type>http://www.w3.org/Protocols/rfc2616-sect10.html#sec10.4.5</ns2:type><ns2:title>Not Found</ns2:title><ns2:errorCode>404</ns2:errorCode><ns2:errorMessage>&lt;![CDATA[InboundJaxrsResponse[context=ClientResponse[method=GET, uri=http://132.145.45.251:5111/RESTMockService/TestUtil/filenames/TS1, status=404, reason=]]]&gt;</ns2:errorMessage><ns2:instance>&lt;![CDATA[("timestamp":1654356175509,"status":404,"error":"Not Found","exception":"com.oracle.cloud.ics.mock.rest.exceptions.CodeNotFoundException","message":"No such country code","path":"/RESTMockService/TestUtil/filenames/TS1")].The HTTP 404, 404 Not Found, and 404 error message is a Hypertext Transfer Protocol (HTTP) standard response code, in computer network communications, to indicate that the client was able to communicate with a given server, but the server could not find the resource that was requested. Carefully re-examine the target endpoint that is being called. ]]&gt;</ns2:instance></ns2:errorCode></ns2:APIInvocationError>

Detailed Error Message
oracle.cloud.connector.api.RemoteApplicationException: Fault Details :
<ns2:APIInvocationError xmlns:ns2="http://xmlns.oracle.com/cloud/generic/rest/fault/REST/getFilename"><ns2:type/><ns2:title/><ns2:detail/><ns2:errorCode/><ns2:errorDetails/><ns2:type>http://www.w3.org/Protocols/rfc2616-sect10.html#sec10.4.5</ns2:type><ns2:title>Not Found</ns2:title><ns2:errorCode>404</ns2:errorCode><ns2:errorMessage>&lt;![CDATA[InboundJaxrsResponse[context=ClientResponse[method=GET, uri=http://132.145.45.251:5111/RESTMockService/TestUtil/filenames/TS1, status=404, reason=]]]&gt;</ns2:errorMessage><ns2:instance>&lt;![CDATA[("timestamp":1654356175509,"status":404,"error":"Not Found","exception":"com.oracle.cloud.ics.mock.rest.exceptions.CodeNotFoundException","message":"No such country code","path":"/RESTMockService/TestUtil/filenames/TS1")].The HTTP 404, 404 Not Found, and 404 error message is a Hypertext Transfer Protocol (HTTP) standard response code, in computer network communications, to indicate that the client was able to communicate with a given server, but the server could not find the resource that was requested. Carefully re-examine the target endpoint that is being called. ]]&gt;</ns2:instance></ns2:errorCode></ns2:APIInvocationError>

```

View the Flow of the Failed Instance and Payload Errors in the Activity Stream on the Instance Details Page

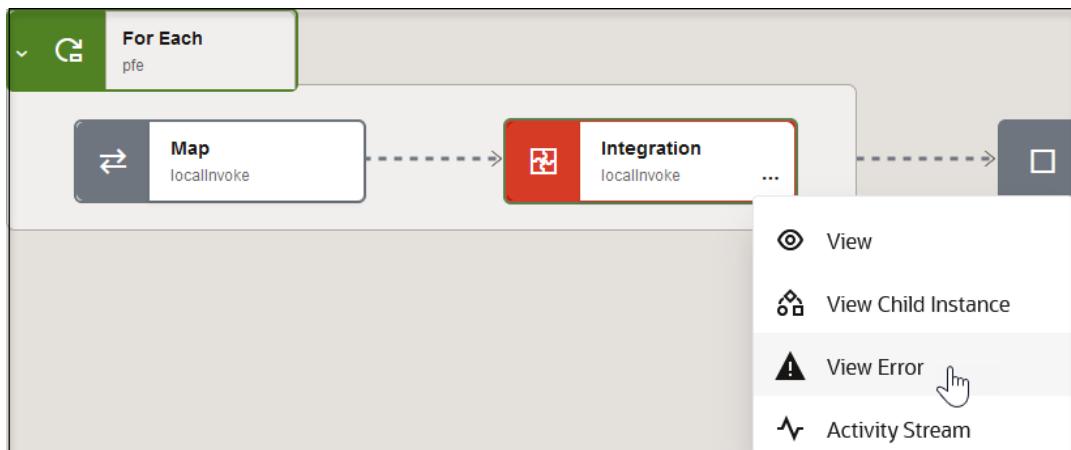
You can view the flow of the failed instance to identify any trigger, actions, or invokes in error on the Instance Details page. You can also drill into those elements to view specific error messages and payload errors.

1. Click a specific business identifier of an integration instance in the **Primary Identifier** column.

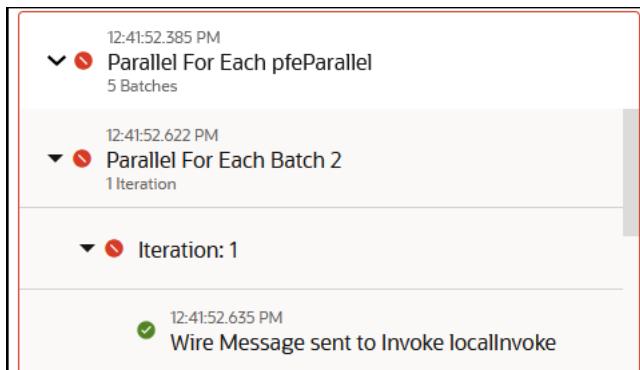


The integration flow of the failed instance and Activity Stream panel are displayed.

2. Scroll through the flow to find the trigger, action, or invoke in error.
3. Click the element, then click **Actions ...** and select **View Error**.



The Activity Stream panel opens to the specific section at which the error occurred. For this example, the error occurred in an integration action (parent integration calling a child integration) of a parallel for-each action. All five of the iterations of the parallel for-each are displayed for viewing in the activity stream.



4. Scroll through and view the errors.
5. Click **Activity Stream**



to close the activity stream. This link operates as a toggle for closing or opening the activity stream.

6. Click the failed trigger, action, or invoke again, then click **Actions ...** and select **Activity Stream**.
This also opens the Activity Stream panel to the specific section at which the error occurred.



7. Scroll through the payload and expand the sections in error to view specific details.
8. Click **View Payload**  to view the payload of the section in error.

02:11:52.894 PM

 Error processing message in Invoke locallnvoke 12ms

Summary	Details			
----------------	----------------	-----------------------------------------------------------------------------------	------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

```

oracle.cloud.connector.api.RemoteApplicationException:
Fault Details :
<ns2:APIInvocationFault
  xmlns:ns2="http://xmlns.oracle.com/cloud/generic
  /rest/fault/collocatedics/localInvoke">
  <ns2:type />
  <ns2:title />
  <ns2:detail />
  <ns2:errorCode />
<ns2:errorDetails>
  <ns2:type>http://www.w3.org/Protocols/rfc2616
  /rfc2616-sec10.html#sec10.4.5</ns2:type>
  <ns2:title>Not Found</ns2:title>
  <ns2:errorCode>404</ns2:errorCode>
<ns2:errorPath>&lt;! [CDATA[InboundJaxrsResponse{context=ClientResponse{method
=POST, uri=http://rest-adapter:8080/ic/api/integration
/v1/flows/rest/LOCALINVOKE1_DLD/1.0/test?message=c1-
invokeMessage-001601, status=404, reason=Not
Found}}]]&gt;</ns2:errorPath>
<ns2:instance>&lt;! [CDATA[Please check the URL and
  ]]&gt;
```

9. Perform other tasks within this section.

Element	Description
Summary	Click to view a summary of payload details. This action also highlights the section of the integration flow in error.
Details	Click to view more specific payload details. This action also highlights the section of the integration flow in error.
Download Payload 	Click to download this part of the payload to your desktop.
Copy Payload 	Click to copy this part of the payload for pasting into a document.

Element	Description
Toggle Line Numbers 	Click to show or hide line numbers in the payload.

10. Click Activity Stream

to close the activity stream.

Resubmit Failed Messages

You can manually resubmit failed messages. Oracle Integration does not automatically resubmit failed messages. After resubmitting an instance, you can track all resubmissions of that instance on the Instances page. You can also change the tracing level for a resubmitted message.

After resubmitting a failed message, the instance can restart from different places in the integration:

- From the beginning of the integration.
- From another location in the integration, depending upon the design. For example, if there is a wait activity in the integration prior to the recoverable fault, then resubmitting the fault starts the integration just after the wait activity.

Faulted instances in asynchronous flows in Oracle Integration are recoverable and can be resubmitted. An exception to this are scheduled flows with scheduled parameters, which are nonrecoverable. Synchronous flows also cannot be resubmitted. You can resubmit an instance a maximum of ten times. You can resubmit errors in the following ways:

- Single failed message resubmissions
- Bulk failed message resubmissions

There is a time window for recovering a failed integration instance that's recoverable. See Service Limits in *Provisioning and Administering Oracle Integration 3*.

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 resubmissions of this error instance are not permitted.

 **Note:**

Do not discard a message that you want to resubmit. A discarded message cannot be resubmitted.

- In the navigation pane, click **Observability**, then **Errors**.
- Click **Filter** , then select the time period or retention period during which to search for integration errors.

Errors			
Time window Last 32 days		Connection Quick_Books_A (QUICK_BOOKS_A)	Display instances All
Mar 13, 2023, 04:45:56 PM GMT		Abort	Retry
17 Errors			
Primary identifier	Instance Id	Fault location	Error time
inputString: 22 QA_QB_SparseUpdateC... 1.0.0	9gaaGa0rEe2CcaN7mkM...	Failed at Sparse QuickBooks	Feb 15, 2023 12:26:24 PM GMT
inputString: 23 QA_QB_SparseUpdateC... 1.0.0	789qV6OrEe2zv_srlFCOg	Failed at Sparse QuickBooks	Feb 15, 2023 12:26:15 PM GMT
SyncToken: 3 QA_QB_CreateCompany... 1.0.0	IZ3cP60hEe2CcaN7mkMM...	Failed at update QuickBooks	Feb 15, 2023 11:12:08 AM GMT
PayType: Check QA_QB_CreateBillPay... 1.0.0	zfujUaz8Ee2zv_srlFCOg	Failed at Create QuickBooks	Feb 15, 2023 06:48:51 AM GMT
inputString: 850 QA_QB_DeleteBill_A 1.0.0	FCLBH6z3Ee2zv_srlFCOg	Failed at Q QuickBooks	Feb 15, 2023 06:07:51 AM GMT

3. Resubmit errors in either of two ways:

- a. Hover over the row of the specific error to resubmit.

- b. Click **Resubmit** .

The Resubmit instance panel opens.

- c. Select the tracing level to use for the resubmitted instance. When resubmitting an errored integration instance, you can change the level of tracing (production, audit, or debug) for just this specific resubmitted instance. Increasing the tracing level (for example, setting it to **Debug**) enables you to view more specific information about errors in the activity stream and downloaded logs. This setting applies only to this instance and does *not* change the tracing level set when the integration was activated.

- **Production**
- **Audit**
- **Debug**

The tracing options provide different levels of information logging. See [Activate an Integration](#).

When the instance resubmission completes, the activity stream provides details. The tracing level shown at the top represents the level set during integration activation. To see details about the tracing level set for this specific resubmission, scroll down to this message:

Recovery process initiated (Tracing level: Debug)

Activity stream

Instance ID: xV6E0CWTee6DiZ1MF6xAeA 

Recoverable
Error in reply to eventTrigger

Tracing level: **Production**   

 Thu Feb 08 02:47:27.299 PM PST 2024, Started by l2serviceadmin	Wire Message dequeued and processed by Trigger eventTrigger
 11:21:49.690 PM, 544ms	Message started processing
 11:21:50.234 PM, 2ms	Data Mapping completed
 11:21:50.236 PM, 41ms	Message sent to reply
  11:21:50.277 PM	Error in reply to eventTrigger
 Thu Jul 20 02:19:43.589 AM IST 2023	Instance resubmitted and queued for recovery (Resubmit count: 1)
 02:19:43.609 AM, 133ms	Recovery process initiated (Tracing level: Debug)
  02:19:43.742 AM, 3ms	Message received by Data Mapper

Or

- Select the check boxes of errors to resubmit together. Only instances identified by the  icon that is visible when you hover over a row can be resubmitted.
- Click **Resubmit** in the upper right corner.

 **Note:**

This method of message resubmission does not allow you to select the tracing level to use for the resubmitted instance. Instead, the tracing level selected during integration activation is used.

- Click **Confirm** to resubmit the instance when prompted.
A Resubmission summary panel opens with details about the instance being resubmitted for processing. Click **Refresh**  to see if it still appears on the Errors page before trying another resubmission.

Instances aren't accepted for resubmission for the following reasons:

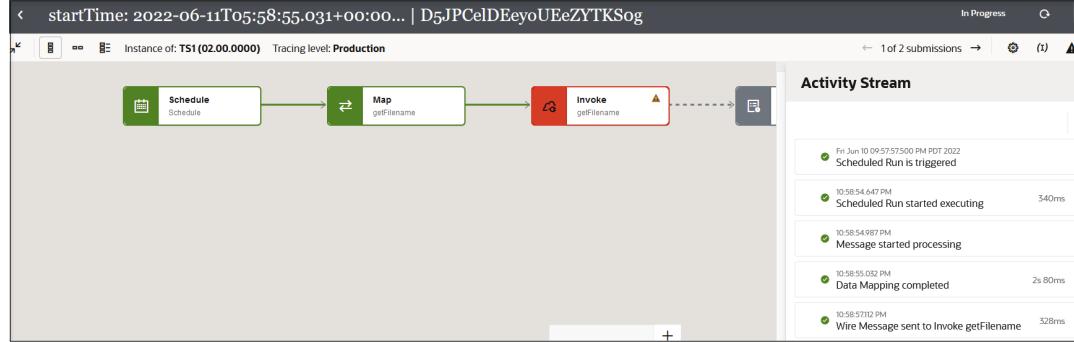
- The instance is not recoverable.
- The instance has reached the maximum number of retries limit.

- The instance has not yet reached the deadline for recovery set internally for batch processing.

You can track all instance runs (both the initial run and all resubmissions) above the activity stream on the Instance Details page.

- In the navigation pane, click **Observability**, then **Instances**.
- Click the instance name.
- Note the number of instance runs above the Activity Stream panel. The number 1 indicates that this is the first run of the integration.

<- 1 of 2 submissions ->



- Click **->** to access tracking details for the first resubmission of this instance (that is, the second run of this instance).

The status for the first resubmitted instance is displayed.



- Click the **<-** to return to the tracking details for the first run of this instance.

Discard Errors

You can discard errors based on the integration in which they occurred. A discarded error message is removed from the Errors page and can be seen in an aborted state on the Instances 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:

- In the navigation pane, click **Observability**, then **Errors**.

- Click **Filter**, then select the time period or retention period during which to search for integration errors.

You can also specify a custom time range for which to search for errors.

Primary identifier	Instance Id	Fault location	Error time
Party Id: 300000084596168 erp_event_trig 1.0.0	XY825sAbEe2M2OMDCx4...	Failed at Return eventTrigger	Mar 11, 2023 02:45:28 PM GMT
Party Id: 300000084596136 erp_event_trig 1.0.0	PyXHo8AVEe2M2OMDCx...	Failed at Return eventTrigger	Mar 11, 2023 02:01:42 PM GMT
Party Id: 300000084597017 erp_event_trig 1.0.0	3Bvg4cAEEe2M2OMDCx4...	Failed at Return eventTrigger	Mar 11, 2023 12:04:23 PM GMT
Party Id: 300000084597004 erp_event_trig 1.0.0	JF4VF7_9Ee29Y49Had1Dvw	Failed at Return eventTrigger	Mar 11, 2023 11:09:07 AM GMT
Party Id: 300000084585988 erp_event_trig 1.0.0	YRFGib_4Ee29Y49Had1Dvw	Failed at Return eventTrigger	Mar 11, 2023 09:49:05 AM GMT
Party Id: 300000084583996 erp_event_trig 1.0.0	88wBG7_xEe2M2OMDCx4...	Failed at Return eventTrigger	Mar 11, 2023 09:49:05 AM GMT

- Discard errors in either of two ways:

a. Select the check boxes of individual errors to discard. Only instances identified by the **Abort** icon that is visible when you hover over a row can be discarded.

b. Click **Abort**.

or

a. Go to the row of the specific error to discard.

b. Click **Abort** .

- Click **Confirm** to abort the instance when prompted.

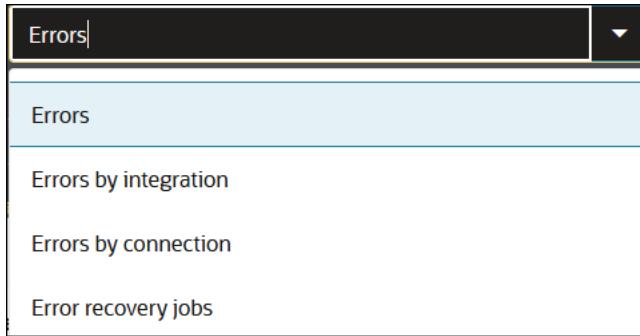
A message is displayed at the top of the page:

Instance ID "number" is submitted for abort. Click on the refresh button to see if this instance still appears on the errors list before trying to abort again.

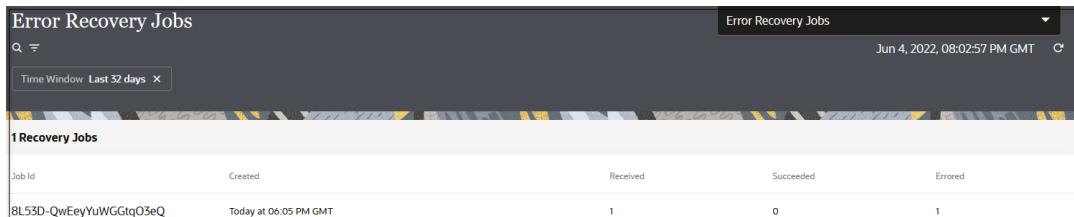
View the Status of Message Recovery

You can search for and view the status of failed messages that have been submitted for recovery on the Errors page.

- In the navigation pane, click **Observability**, then **Errors**.
- Submit a failed message for recovery.
- From the drop-down list at the top right, select **Error Recovery Jobs**.



4. The Error Recovery Jobs page is displayed for viewing the status of message recovery.
 - **Received:** Recovery is still in progress.
 - **Succeeded:** Recovery succeeded.
 - **Errored:** Recovery failed.

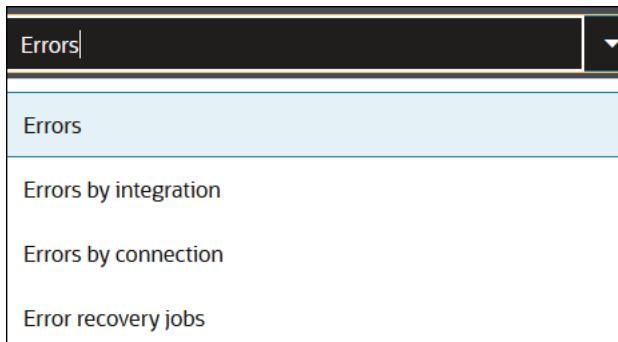


Job Id	Created	Received	Succeeded	Errored
8L53D-QwEeyYuWGtqO3eQ	Today at 06:05 PM GMT	1	0	1

View Errors By Integration and Connection

You can view errors by integration and connection on the Errors page.

1. In the navigation pane, click **Observability**, then **Errors**.
2. From the drop-down list at the top, select an option:
 - To view errors by integration, select **Errors by integration**.
 - To view errors by connection, select **Errors by connection**.



3. If you selected **Errors by integration** or **Errors by connection**, click **Filter**  to show errors alphabetically by name or by error count in descending order.

4. Click the error count for an integration or connection to invoke the Errors page and view all errored-out instances for the given integration or connection.
5. Hover over an integration:

- Click **Retry**  to resubmit errors.
- Click **Abort**  to discard errors.

View the Business Identifiers Used in Failed Integration Instances

You can view the business identifiers used in failed integration instances.

1. In the navigation pane, click **Observability**, then **Errors**.
2. Click the business identifier of the failed integration instance.

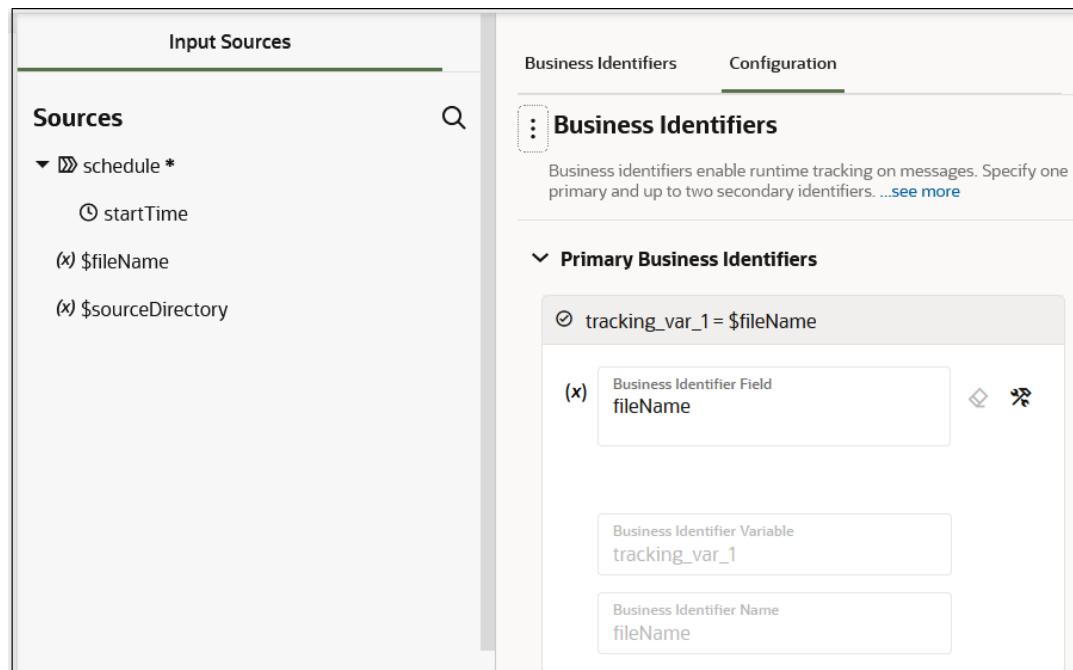


The integration instance and the Activity Stream panel are displayed.

3. Click **Business Identifiers**  above the activity stream to display all the defined business identifiers and values for the integration.

Business Identifiers	Configuration
Name	Value
 fileName	invoke 2000.csv

4. Click **Configuration**.
5. Click **Actions**  to show the input sources for the business identifiers.



See [Assign Business Identifiers](#) and [Track Integration Instances](#).

Track Integration Instances

You can filter and track the status of integration instances on the Instances page through business identifiers. You can also view the message payload of an instance you are tracking.

Topics:

- [Filter the Display of Integration Instances](#)
- [Search for Business Identifier Values](#)
- [Track the Status of Integration Instances](#)
- [Determine Why an Integration Instance Stopped Running](#)
- [Replay Integration Instances](#)

You assign business identifiers during integration design time. At least one business identifier is required. See [Assign Business Identifiers for Tracking Fields in Messages](#).

Filter the Display of Integration Instances

You can filter the display of integration instances on the Instances page during runtime.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Click **Filter** to filter the display of instances by the following:
 - Last updated or creation time
 - Time window
 - Instance ID
 - Status

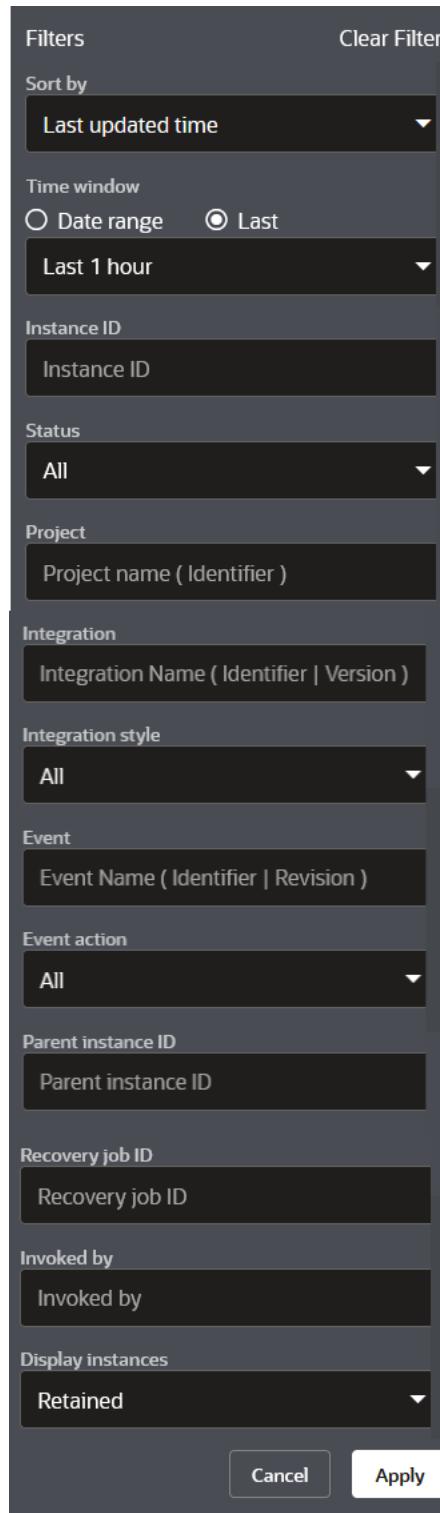
- Project name
- Integration name

Begin typing an integration name. Do not use copy and paste. The filter is refreshed to display fields for the defined secondary business identifiers.

You can now begin typing the values of the business identifiers to narrow down the search. This input follows the same rules (for example, for exact matches and phrase matches) as described in [Search for Primary and Secondary Business Identifier Values](#).

- Integration style (all, application, schedule, or event)
- Event name
- Instances published and/or subscribed to that event name
- Parent instance ID
- Recovery job ID
- Integration that invoked the instance
- Instance state (all, retained, purged, replayed, or replayable).

You can enter variable values with special characters or variable values of multiple words with white space in double quotes in the **Search**  field (for example, &filename or "file tracking").



- Click **Date Range** to specify a custom time range with which to filter the search for instances with the **From** and **To** buttons. This enables you to select a time range as small as a minute and track only those instances processed within that specified time range. The specified time range is preserved by default and applied to the data being displayed when you navigate from one monitoring page to another (for example, when you move from the Instances page to the Monitor Integrations page or Errors page).

 **Note:**

The time range specified must be within the retention period. Data beyond the retention period is not retained. If the selected time range is beyond the retention period, only available data is displayed.

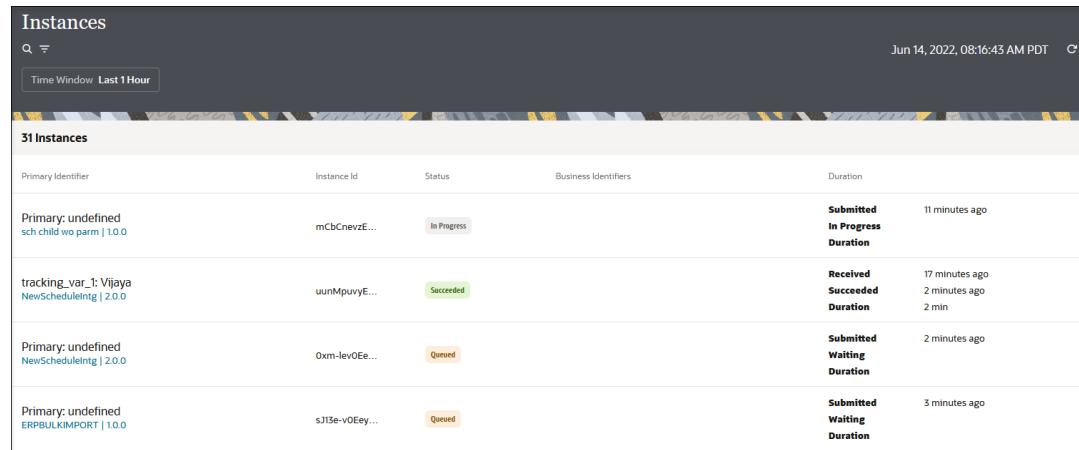
Search for Business Identifier Values

You can search for primary and secondary business identifier values on the Instances page during runtime. You can perform simple searches, phrase match searches, and exact match searches. This section provides examples of performing each search type.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Use the **Search** field to search for values across both primary and secondary business identifiers.

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). You can show the name and value of tracking variables.

For canceled instances, the running time does *not* point to the actual time the canceled (faulted) instance was running. Instead, it consists of the total time span between when the instance began and when the instance was manually canceled.



Instances				
<input type="text"/> <input type="button" value="="/> <input type="button" value="Time Window"/> <input type="button" value="Last 1 Hour"/>				
31 Instances				
Primary Identifier	Instance Id	Status	Business Identifiers	Duration
Primary: undefined sch child wo parm 1.0.0	mCbCnevzE...	In Progress		Submitted In Progress Duration 11 minutes ago
tracking_var_1: Vijaya NewScheduleIntg 2.0.0	uunMpuvyE...	Succeeded		Received Succeeded Duration 17 minutes ago 2 minutes ago 2 min
Primary: undefined NewScheduleIntg 2.0.0	0xm-lev0Ee...	Queued		Submitted Waiting Duration 2 minutes ago
Primary: undefined ERPBULKIMPORT 1.0.0	sJ13e-v0Eey...	Queued		Submitted Waiting Duration 3 minutes ago

The **Search** field provides many capabilities:

- [Search for Primary and Secondary Business Identifier Values](#)
- [Perform a Simple Search](#)
- [Perform a Phrase Match Search](#)
- [Perform an Exact Match Search](#)

 **Note:**

- Partial word searches are not supported. For example, assume one business identifier value is `New York` and another is `New Orleans` and both belong to different instance IDs. If you enter `New` as your search criterion, both instances are displayed because `New` and `York` are considered separate words. White space is considered a word separator.
- Underscores are not considered word separators. For example if the values are `New_York` and `New_Olleans`, and you enter `New` as the search criterion, neither instance is displayed because `New_York` is considered a complete word and partial word matches/searches are not supported. This is the case only with underscores. All other special characters (for example, periods, commas, and others) are considered word separators.

Search for Primary and Secondary Business Identifier Values

1. Enter a value, and press **Enter**. For example, assume the primary value is `James`. In the search results, any instances that have a primary value of **James** are returned. Any instances with a secondary value that includes **James** (for example, `Mark.James@asc.com`) are also returned.
2. Search only for primary business identifier values by entering the value as `primary: value` or `Primary: value` in the **Search** field.

primary: James

 **Note:**

- If you enter a primary business identifier in the **Search** field, but do not press **Enter**, then select a value from the time period drop-down list, note that the instances are filtered considering the string entered in the **Search** field, even though **Enter** was not pressed. This is the expected behavior and is true for other landing pages in Oracle Integration.
- The search facility on the Instances page is case sensitive.

Perform a Simple Search

1. Enter a value in the **Search** field, and press **Enter**. For example:

125-78-2345

This returns all instances that include part of that business identifier value.

The screenshot shows the 'Instances' page in the Oracle Integration Cloud Service. A search bar at the top contains the value '125-78-2345'. Below the search bar, there are filter buttons for 'Time window Last 1 hour' and 'Integration Schedule Parameters Sample (SCHEDULE_PARAMETERS_SAMPLE | 01.00.0000)'. The main area displays a table titled '6 instances' with columns: Primary identifier, Instance ID, Status, and Business Identifiers.

Primary identifier	Instance ID	Status	Business Identifiers
area: 125-78-2345 New York US Schedule Parameters... 1.0.0	pcZC7GJ6E...	Succeeded	position 0 tracking_v... 0 files
area: 125-78-2345 Delhi India Schedule Parameters... 1.0.0	mUgr7GJ6...	Succeeded	position 0 tracking_v... 0 files
area: 125-78-2345 Delhi Schedule Parameters... 1.0.0	ZrtYFmJ6E...	Succeeded	position 0 tracking_v... 0 files
area: 125-78-2345 York Schedule Parameters... 1.0.0	MVQUf2J6...	Succeeded	position 0 tracking_v... 0 files
area: 125-78-2345 York Schedule Parameters... 1.0.0	LAvZC2J6E...	Succeeded	position 0 tracking_v... 0 files
area: 124-76-2345 York Schedule Parameters... 1.0.0	C1O16GJ6E...	Succeeded	position 0 tracking_v... 0 files

2. Enter a value with white space in the **Search** field, and press **Enter**. For example:

125-78-2345 York

Note that each individual part of the value is treated as a separate value (125, 78, 2345, and York are all considered separate). Any business identifier value found containing at least one of these values is returned. A simple search is similar to a web browser search.

Instances			
<input type="text" value="125-78-2345 York"/> x =			
Primary identifier	Instance ID	Status	Business Identifiers
area: 125-78-2345 New York US Schedule Parameters... 1.0.0	pcZC7GJ6E...	Succeeded	position 0 tracking_v... 0 files
area: 125-78-2345 Delhi India Schedule Parameters... 1.0.0	mUgr7GJ6...	Succeeded	position 0 tracking_v... 0 files
area: 125-78-2345 Delhi Schedule Parameters... 1.0.0	ZrtYFmJ6E...	Succeeded	position 0 tracking_v... 0 files
area: 125-78-2345 York Schedule Parameters... 1.0.0	MVQUf2J6...	Succeeded	position 0 tracking_v... 0 files
area: 125-78-2345 York Schedule Parameters... 1.0.0	LAvZC2J6E...	Succeeded	position 0 tracking_v... 0 files
area: 124-76-2345 York Schedule Parameters... 1.0.0	C1O16GJ6E...	Succeeded	position 0 tracking_v... 0 files
area: New York Schedule Parameters... 1.0.0	WPovj2J5E...	Succeeded	position 0 tracking_v... 0 files

Perform a Phrase Match Search

You can perform a phrase match search by enclosing your value in double quotes. Phrase matches are useful when you need to search for primary and secondary business identifier values containing special characters and white spaces.

1. Enter a value in the **Search** field, and press **Enter**. For example:

"125-78-2345 York"

2. Note that exact values are returned. Without the double quotes, each individual part would be considered a separate searchable value: 125, 78, 2345, and York.

Instances															
"125-78-2345 York"															
Time window Last 1 hour Integration Schedule Parameters Sample (SCHEDULE_PARAMETERS_SAMPLE 01.00.0000) X Display instances Retained															
2 instances															
<table border="1"> <thead> <tr> <th>Primary identifier</th><th>Instance ID</th><th>Status</th><th>Business Identifiers</th></tr> </thead> <tbody> <tr> <td>area: 125-78-2345 York Schedule Parameters... 1.0.0</td><td>MVQUf2J6...</td><td>Succeeded</td><td>position 0 tracking_v... 0 files</td></tr> <tr> <td>area: 125-78-2345 York Schedule Parameters... 1.0.0</td><td>LAvZC2J6E...</td><td>Succeeded</td><td>position 0 tracking_v... 0 files</td></tr> </tbody> </table>				Primary identifier	Instance ID	Status	Business Identifiers	area: 125-78-2345 York Schedule Parameters... 1.0.0	MVQUf2J6...	Succeeded	position 0 tracking_v... 0 files	area: 125-78-2345 York Schedule Parameters... 1.0.0	LAvZC2J6E...	Succeeded	position 0 tracking_v... 0 files
Primary identifier	Instance ID	Status	Business Identifiers												
area: 125-78-2345 York Schedule Parameters... 1.0.0	MVQUf2J6...	Succeeded	position 0 tracking_v... 0 files												
area: 125-78-2345 York Schedule Parameters... 1.0.0	LAvZC2J6E...	Succeeded	position 0 tracking_v... 0 files												

Note that exact phrase match searches can also return the exact match, plus any additional values at the beginning or end of the enclosed value. Consider the following example.

- Enter a value in the **Search** field, and press **Enter**. For example:

"125-78-2345 Delhi"

This returns any value that matches, plus any additional values. In this case, both 125-78-2345 Delhi and 125-78-2345 Delhi India are returned.

Instances															
"125-78-2345 Delhi"															
Time window Last 1 hour Integration Schedule Parameters Sample (SCHEDULE_PARAMETERS_SAMPLE 01.00.0000) X Disp															
2 instances															
<table border="1"> <thead> <tr> <th>Primary identifier</th><th>Instance ID</th><th>Status</th><th>Business Identifiers</th></tr> </thead> <tbody> <tr> <td>area: 125-78-2345 Delhi India Schedule Parameters... 1.0.0</td><td>mUgr7GJ6...</td><td>Succeeded</td><td>position 0 tracking_v... 0 files</td></tr> <tr> <td>area: 125-78-2345 Delhi Schedule Parameters... 1.0.0</td><td>ZrtYFmJ6E...</td><td>Succeeded</td><td>position 0 tracking_v... 0 files</td></tr> </tbody> </table>				Primary identifier	Instance ID	Status	Business Identifiers	area: 125-78-2345 Delhi India Schedule Parameters... 1.0.0	mUgr7GJ6...	Succeeded	position 0 tracking_v... 0 files	area: 125-78-2345 Delhi Schedule Parameters... 1.0.0	ZrtYFmJ6E...	Succeeded	position 0 tracking_v... 0 files
Primary identifier	Instance ID	Status	Business Identifiers												
area: 125-78-2345 Delhi India Schedule Parameters... 1.0.0	mUgr7GJ6...	Succeeded	position 0 tracking_v... 0 files												
area: 125-78-2345 Delhi Schedule Parameters... 1.0.0	ZrtYFmJ6E...	Succeeded	position 0 tracking_v... 0 files												

Perform an Exact Match Search

You can return an exact match search by enclosing the business identifier value in brackets ([and]). This search returns only exact match values.

- Enter a value in brackets in the **Search** field, and press **Enter**. For example:

[125-78-2345 Delhi]

- View the results. Only exact matches are returned.

The screenshot shows the 'Instances' page with a search bar containing '[125-78-2345 Delhi]'. Below the search bar are two buttons: 'Time window Last 1 hour' and 'Integration Schedule Parameters Sample (SCHEDULE_PARAMETERS_SAMPLE | 01.00.0000)'. A close button 'x' is also present. The main content area displays a table with one row, labeled '1 instances'. The table has four columns: 'Primary identifier', 'Instance ID', 'Status', and 'Business Identifiers'. The 'Primary identifier' column contains 'area: 125-78-2345 Delhi Schedule Parameters... | 1.0.0'. The 'Instance ID' column contains 'ZrtYFmJ6E...'. The 'Status' column is green and says 'Succeeded'. The 'Business Identifiers' column contains 'position 0 tracking_v... 0 files'.

Track the Status of Integration Instances

You can track fields in messages on which you have defined business identifiers on the Instances page during runtime. These fields are available for tracking on the Instances page because you defined at least one business identifier in the Business Identifiers panel during design time.

1. In the navigation pane, click **Observability**, then **Instances**.
2. Perform the following tasks:
 - [View the Slowest Running Activities in an Integration Instance](#)
 - [Discard an Instance on the Instances Page](#)
 - [View the Status of Future Schedule Integrations](#)
 - [View the Activity Stream on the Instances Page](#)
 - [View the Child Instances of a Parent Instance](#)
 - [View the Activity Stream of a Child Integration Separately from the Parent Integration](#)
 - [View the Display of HTTP Headers Separate from the Payload](#)
 - [View File Metadata for Large Payloads in the Activity Stream](#)
 - [View the Activity Stream for Adapters Created with the Rapid Adapter Builder](#)

You can also view the activity stream, business identifiers and values, and message payload in failed instances. See [Manage Errors](#).

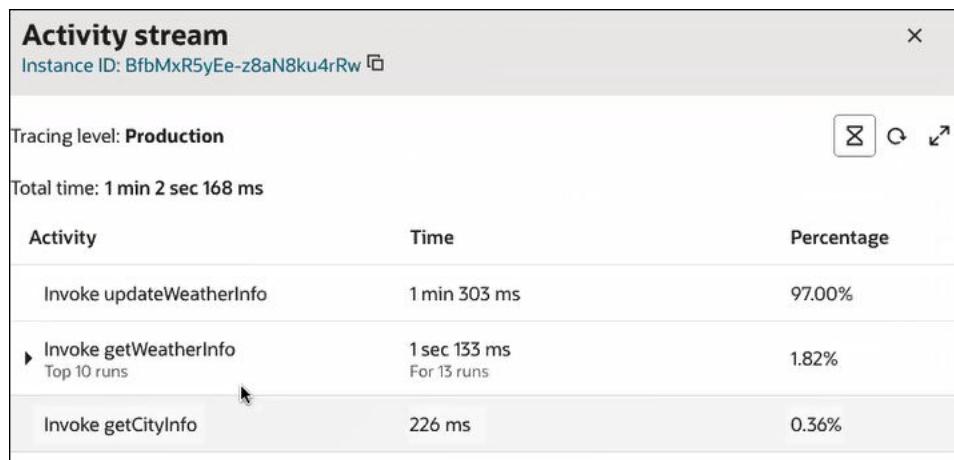
View the Slowest Running Activities in an Integration Instance

To diagnose performance issues, you can view the activities in the activity stream by their elapsed time. This ability ensures that the most time-consuming activities show first in the list and the best performing activities appear at the bottom. This functionality provides you with an immediate performance view and eliminates the time-consuming task of clicking each individual message or a large number of iterations in a looping action to identify time-consuming activities.

Understand the following behavior:

- The duration is *not* shown for all activities. The duration for an activity is the time from the beginning to the end of activity execution. Not all activities have both times (for example, a trigger or a throw action) because both are not applicable.

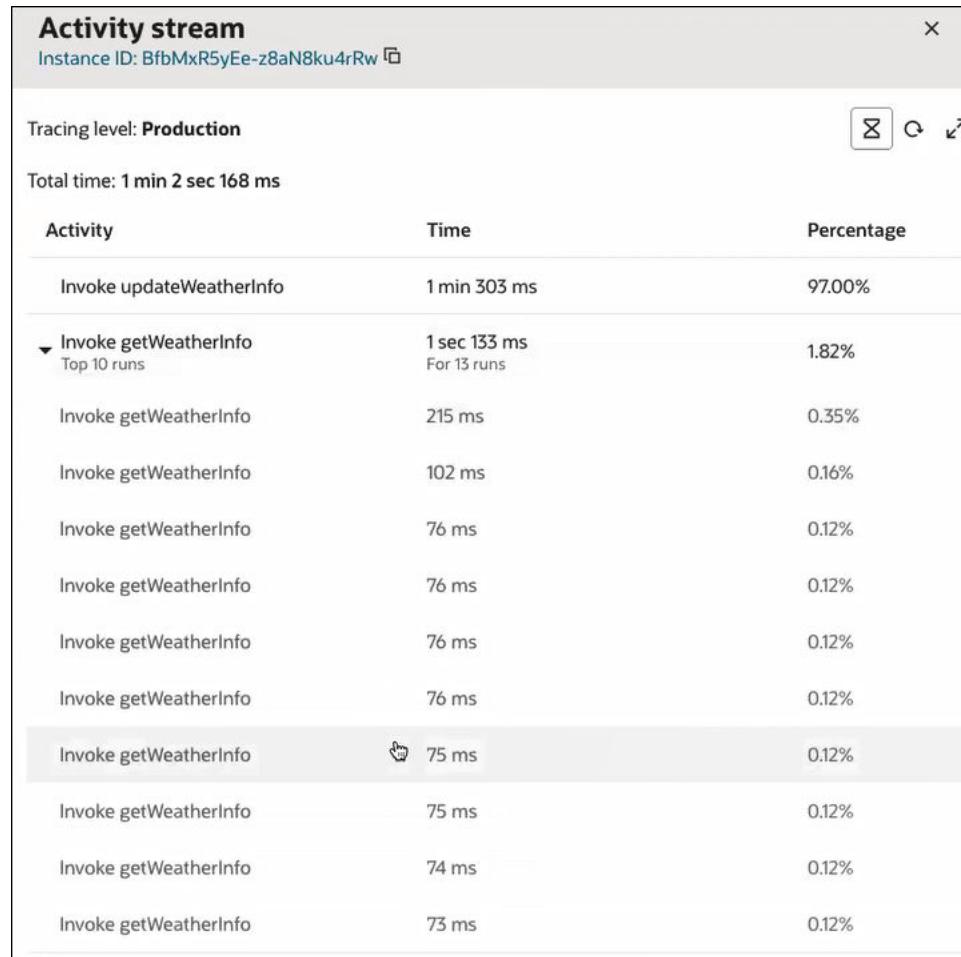
- The duration may not be available even for activities that have both times (such as a map action). This is because availability of these times depends on the integration tracing level. For example, an invoke action always appears, but other actions appear only when the instance is executed with the tracing level set to debug.
 - Only the top 10 slowest activities are shown in the table. Your integration may have many more actions.
 - You can view the overall time a looping action (for example, a while action or a for-each action) took to execute all its iterations. You can also expand the looping action to view the following:
 - If your looping action had less than 10 iterations, all iterations and their times to execute are shown.
 - If your looping action has more than 10 iterations, only the ten slowest iterations and their times to execute are shown.
- Go to the row of the integration instance to view. The **Duration** column shows the time that an instance took to execute.
 - Click **View Details** .
- The activity stream is displayed.
- Click **View slowest activities** .
- The total duration of the instance is displayed at the top. The activities that took the longest amount of time to execute are displayed in a table. The final column shows the percentage of time each activity took to execute based on the total duration of the instance. For this example, **getWeatherInfo** included a looping action that iterated 13 times.



The screenshot shows the 'Activity stream' interface for an integration instance. The instance ID is BfbMxR5yEe-z8aN8ku4rRw. The tracing level is set to Production. The total execution time is 1 min 2 sec 168 ms. A table lists the slowest activities:

Activity	Time	Percentage
Invoke updateWeatherInfo	1 min 303 ms	97.00%
▶ Invoke getWeatherInfo Top 10 runs	1 sec 133 ms For 13 runs	1.82%
Invoke getCityInfo	226 ms	0.36%

- Expand the looping action (**getWeatherInfo**) to view the iterations. For this example, the looping action was executed 13 times. The 10 slowest-running iterations are shown.



5. Use this functionality to identify and debug slow running activities.
6. Click **View slowest activities** ✖ again to return to the activity stream. This link enables you to toggle back and forth.

Discard an Instance on the Instances Page

1. Hover over the row of the instance to discard, then select **Abort** ✖.



This option is only visible under the following scenarios:

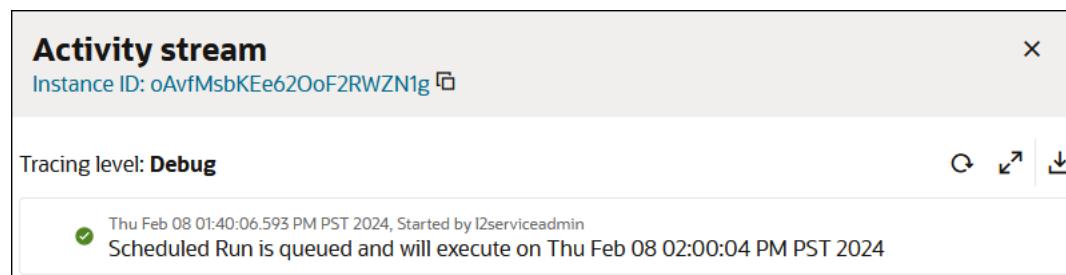
- In-progress instances of asynchronous integrations (both scheduled and nonscheduled).
- Waiting and paused instances of schedule integrations. Attempting to discard other types of instances results in an error message being displayed.

View the Status of Future Schedule Integrations

Schedule integrations that have not yet run are displayed as **Queued** on the Instances page. The **Duration** column shows the label **Scheduled on** and the time at which the queued integration is scheduled to run (for this example, in four minutes).

Primary Identifier	Instance Id	Status	Business Identifiers	Duration
Primary: undefined TestSchedule 1.0.0	1Sg6-fPcEey...	Queued		Submitted Scheduled on Duration 6 minutes ago in 4 minutes

1. Hover over the queued integration in the **Duration** column to display the full date and time at which the queued integration is scheduled to run.
2. Click **View Details**  or click the business identifier in the **Primary Identifier** column of a queued integration. The Activity Stream panel opens in either place and shows the time at which the queued integration is scheduled to run.



The screenshot shows the Activity stream panel for an instance. It includes the title "Activity stream", the instance ID "oAvfMsbKEe62OoF2RWZN1g", tracing level "Debug", and a message indicating a scheduled run: "Thu Feb 08 01:40:06.593 PM PST 2024, Started by l2serviceadmin" and "Scheduled Run is queued and will execute on Thu Feb 08 02:00:04 PM PST 2024".

View the Activity Stream on the Instances Page

1. Hover over an instance, then select **View Details**  to view the activity stream of the instance.
The activity stream shows details about the movement of the message through the integration, including where any failures occurred. The message flow through each action is also shown.
2. Expand the invoke connection.
Key child messages of the parent invoke connection are displayed.

Activity stream

Instance ID: TeROiMbMEe6jpjEOTjvf1Q [Copy](#)

Tracing level: **Production** [Change](#)

Thu Feb 08 01:52:07.754 PM PST 2024, Started by l2serviceadmin
Scheduled Run is queued

02:01:07.824 PM, 60ms
Scheduled Run started executing

02:01:07.884 PM
Message started processing

02:01:07.912 PM, 1ms
Message received by Assignment assign

02:01:07.913 PM, 71ms
Data Mapping completed

> 02:01:07.984 PM
Invoke InvokeComments

02:01:08.034 PM, 1m 0s 126ms
Wait Wait1 triggered

④ 02:02:08.160 PM
Logger logger

02:02:08.186 PM
Processing completed successfully

The date and time according to your user preferences are displayed. If your instance contains logic actions (for example, a while action or a for-each action with multiple iterations), each iteration is shown in the activity stream and can be expanded to show specific details about the message flow.

08:22:52.407 AM

For Each IterateFilesInZip
4 Iterations

Iteration: 1

Iteration: 2

Iteration: 3

Iteration: 4

3. Click **Download**  at the top of the activity stream to download the activity stream logs for the entire instance.

- 4.** Click **View Payload** to view the payload for that part of the integration flow.

④ 01:52:28.869 PM
In Global Fault Handler - Message sent by Notification

{ "subject": "Integration Name Oracle SAP SuccessFactor EBS Employee Sync Error Notification", "fromAddress": "no-reply@oracle.com", "mimeType": "text/html", "toAddresses": "kp72259@gmail.com", "body": "<html>\n<head>\n<meta http-equiv=\"content-type\" content=\"text/html; charset=windows-1252\">\n</meta>\n<style>\n <table >\n <font-family: arial, sans-serif;\n <font-size: 14px;\n <border-collapse: collapse;\n <width: auto;\n <height: auto;\n </table>\n</style>\n<body>\n</body>\n</html>" }

- ## **5. Perform tasks within the payload.**

Element	Description
Summary	Click to view a summary of payload details.
Details	Click to view more specific payload details.
Download Payload 	Click to download this part of the payload to your desktop.
Copy Payload 	Click to copy this part of the payload for pasting into a document.
Toggle Line Numbers 	Click to show or hide line numbers in the payload.

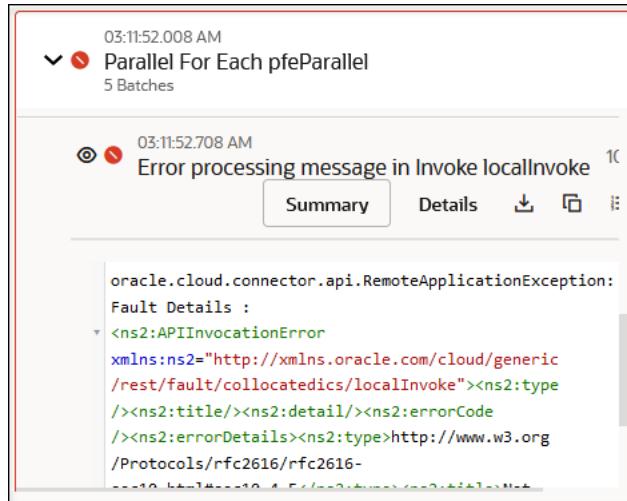
View the Child Instances of a Parent Instance

If your instance invokes child instances (for example, an integration action in your flow is invoking child integrations), you can view those instances on the Instances page.

1. Hover over a row and select **View Child Instances**.

start Time: 2022-06-20T15:32:59.714+00:00	3jB82FcEe...	Succeeded	Received Succeeded Duration	13 minutes ag 3 minutes ag:
TestSchedule 1.0.0				Schedule
Primary: undefined	RAJ0kPCuE...	Queued	Submitted Waiting Duration	View Child Instances
TestSchedule 1.0.0				

You can also view invocations of child instances (local invokes) in the activity stream.

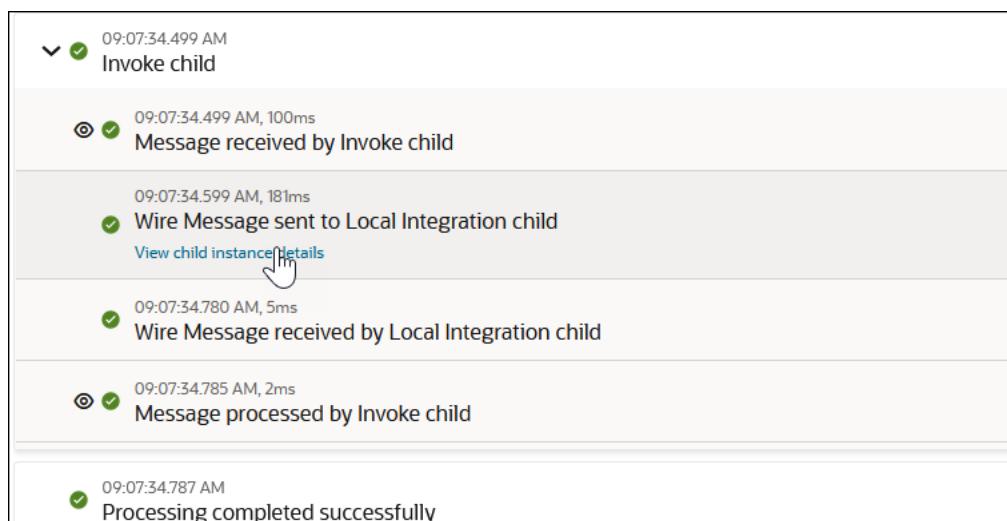


View the Activity Stream of a Child Integration Separately from the Parent Integration

You can view a child integration being invoked by a parent integration in a separate activity stream. You can also view the separate activity streams for one integration that invokes another integration, which then invokes a child integration. Each activity stream is connected by a button that provides easy access.

See [Invoke a Child Integration from a Parent Integration](#).

1. Hover over the instance of a parent integration that invokes a child integration.
2. Click **View Details** . The activity stream is displayed.
3. Expand the invoke connection. The **View child instance details** link is displayed. This is where the instance of the child integration is invoked.



4. Click **View child instance details** to access the instance of the activity stream for the child integration. Note that **Integration: Child** and the integration version number are also displayed after the instance ID at the top.

The screenshot shows the 'Activity stream' interface. At the top, it displays the 'Instance Id: D-ENOG9pEe21yVONUI5uYA' and 'Integration: Child|01.00.0000'. Below this, the 'Tracing level: Debug' is set. The main area lists three processing events:

- A checkmark icon followed by the timestamp 'Tue Nov 29 09:07:34.701 AM MAGT 2022, 54ms' and the message 'Wire Message dequeued and processed by Trigger child'.
- A circled checkmark icon followed by the timestamp '09:07:34.755 AM, 22ms' and the message 'Message started processing'.
- A checkmark icon followed by the timestamp '09:07:34.777 AM' and the message 'Processing completed successfully'.

5. Click to return to the instance of the activity stream for the parent integration.
6. Return to the Instances page and hover over the instance of an integration that invokes another integration, which then invokes a child integration.
7. Click **View Details** .
8. Expand the invoke connection.
9. Click **View child instance details** to access the instance of the activity stream for the integration being invoked.

Activity stream
Instance Id: WPVwf29qEe2HV-eope8Lmg

Tracing level: **Debug**

Tue Nov 29 09:16:46.805 AM MAGT 2022, 208ms
Wire Message dequeued and processed by Trigger createcustomer

09:16:47.013 AM, 25ms
Message started processing

09:16:47.038 AM, 3ms
Message received by Data Mapper

09:16:47.041 AM, 2ms
Data Mapping completed with Message

09:16:47.043 AM
Invoke test

09:16:47.043 AM, 281ms
Message received by Invoke test

09:16:47.324 AM, 214ms
Wire Message sent to Local Integration test
[View child instance details](#)

09:16:47.538 AM, 4ms
Wire Message received by Local Integration test

09:16:47.542 AM, 2ms
Message processed by Invoke test

09:16:47.544 AM
Processing completed successfully

Note that the parent name and the integration version number are displayed at the top.

10. Expand the invoke connection.

The screenshot shows the Activity stream interface with the following details:

Activity stream
Instance Id: WVRZYW9qEe21yVONUI5uYA [🔗](#)
Integration: Parent Async|01.00.0000

Tracing level: **Debug**

The trace log contains the following entries:

- Tue Nov 29 09:16:47.427 AM MAGT 2022, 63ms
Wire Message dequeued and processed by Trigger createcustomer
- ④ ✓ 09:16:47.490 AM, 43ms
Message started processing
- ④ ✓ 09:16:47.533 AM, 3ms
Message received by Data Mapper
- ④ ✓ 09:16:47.536 AM, 1ms
Data Mapping completed with Message
- ▼ ✓ 09:16:47.537 AM
Invoke test
 - ④ ✓ 09:16:47.537 AM, 79ms
Message received by Invoke test
 - 09:16:47.616 AM, 158ms
✓ Wire Message sent to Local Integration test
[View child instance details](#)
 - ④ ✓ 09:16:47.774 AM, 4ms
Wire Message received by Local Integration test
 - ④ ✓ 09:16:47.778 AM, 2ms
Message processed by Invoke test

11. Click **View child instance details** to access the instance of the activity stream for the child integration being invoked.
Note that the child name and the integration version number are displayed at the top.

The screenshot shows the Activity stream interface for an integration instance. At the top, it displays the Instance Id: WX52qW9qEe2HV-eope8Lmg and the Integration: CHILD_TRACKING_DETAILS_TEST|01.00.0000. The tracing level is set to Debug. The activity stream lists the following events:

- Tue Nov 29 09:16:47.703 AM MAGT 2022, 55ms: Wire Message received by Trigger EmployeeAPI
- 09:16:47.758 AM, 2ms: Message started processing
- 09:16:47.760 AM, 3ms: Message received by Data Mapper
- 09:16:47.763 AM, 1ms: Data Mapping completed with Message
- 09:16:47.764 AM, 7ms: Message sent to reply
- 09:16:47.771 AM: Reply to EmployeeAPI completed with Wire Message

At the bottom right of the interface, there are two icons: a double arrow pointing up and down, and a downward arrow.

12. Click twice to return to the instance of the activity stream for the top level integration.

View the Display of HTTP Headers Separate from the Payload

HTTP headers associated with a payload are displayed separately from the payload for ease of viewing in the activity stream.

1. Click **View Details** .

Activity stream

Recoverable
Fault rethrown successfully by Re-throw Fault

Tracing level: **Debug**

- Sat Sep 10 01:01:58.469 AM PDT 2022, 85ms
 - Wire Message dequeued and processed by Trigger receiveOFSCActivity
- 01:01:58.554 AM, 31ms
 - Message started processing
- 01:01:58.585 AM, 0ms
 - Message entered Scope scope_global
- 01:01:58.585 AM, 1ms
 - Message entered Switch
- 01:01:58.586 AM, 0ms
 - Message routed through Switch Route with condition Otherwise

If HTTP headers are associated with the payload, two tabs are displayed: **Payload** and **Headers**.

Activity stream

Recoverable
Fault rethrown successfully by Re-throw Fault

Tracing level: **Debug**

Sat Sep 10 01:01:58.469 AM PDT 2022, 85ms

- Wire Message dequeued and processed by Trigger receiveOFSCActivity

Payload **Headers**

```
{
  "eventType": "activityCreated",
  "time": "2022-09-10 08:00:50",
  "user": "",
  "activityDetails": {
    "date": "2022-10-09",
    "activityId": 4267570,
    "resourceId": "donna_stanley",
    "resourceInternalId": 8100387,
    "apptNumber": ""
  },
  "activityChanges": {
    "teamResourceId": "lv_two_crew",
    "teamResourceInternalId": 8100383,
    "recordType": "regular",
    "status": "pending",
    "activityType": "crew_assignment",
    "duration": 5,
    "language": "en",
    "languageISO": "en-US",
    "timeZone": "Pacific"
  }
}
```

- Click **Headers** to view details about the HTTP headers associated with the payload.

The screenshot shows the Oracle Integration Activity Stream interface. At the top, there is a red banner with the text "Recoverable" and "Fault rethrown successfully by Re-throw Fault". Below this, the tracing level is set to "Debug". A message log entry is displayed, showing a wire message being dequeued and processed by a trigger. The "Headers" tab is selected, revealing the following HTTP headers:

```

HTTP-HEADERS
authorization: *****
accept: application/json
content-type: application/json
user-agent: Jersey/2.35 (HttpURLConnection 17.0.3.1)
content-length: 552

```

Below the headers, another message log entry shows a message starting to process at 01:01:58.554 AM, 31ms.

View File Metadata for Large Payloads in the Activity Stream

You can view file metadata for large payloads in the activity stream.

For example, if an integration performing inbound polling sends a CXML payload in GZIP format, the payload is shown in readable format in the activity stream. The payload size and media type are also shown.

View the Activity Stream for Adapters Created with the Rapid Adapter Builder

When the tracing level is set to **Debug**, you can view messages for the top level invoke connection and any invoke connections it calls. You can expand each invoke connection being called to view messages for preprocessing logs, wire messages, and postprocessing logs.

- Click **View Details** for an instance that includes an adapter created with the Rapid Adapter Builder.
- View the activity stream and note that a single invoke connection (**listMembers**) for an adapter created with the Rapid Adapter Builder is calling multiple invoke connections.

▼ ✓ 01:23:59.867 AM	Invoke listMembers
⌚ ✓ 01:23:59.867 AM	Message received by Invoke listMembers
⌚ ✓ 01:23:59.967 AM	Invoke listMembers processing starts
▶ ✓ 01:23:59.968 AM	Invoke 1
▶ ✓ 01:24:00.220 AM	Invoke 2
⌚ ✓ 01:24:00.337 AM	Invoke listMembers processing ends
⌚ ✓ 01:24:00.360 AM	Message processed by Invoke listMembers

3. Expand an invoke. Each invoke can have preprocessing and postprocessing loggers before and after the invocation. Wire messages sent and received by **listMembers** are also shown.

▼ ✓ 01:23:59.968 AM	Invoke 1
⌚ ✓ 01:23:59.968 AM	Pre processing logger 1
⌚ ✓ 01:23:59.970 AM	Pre processing logger 2
⌚ ✓ 01:23:59.971 AM	Pre processing logger 3
✓ 01:23:59.989 AM, 224ms	Wire Message sent by Invoke listMembers
⌚ ✓ 01:24:00.213 AM	Wire Message received by Invoke listMembers
⌚ ✓ 01:24:00.215 AM	Post processing logger 1
⌚ ✓ 01:24:00.216 AM	Post processing logger 2
⌚ ✓ 01:24:00.218 AM	Post processing logger 3
⌚ ✓ 01:24:00.219 AM	Post processing logger 4

- Click **View Details**  to view details about the payload and the messages being logged.

Track the Integration Flow on the Instance Details Page

The Instance Details page provides a graphical view of the instance flow and access to the activity stream.

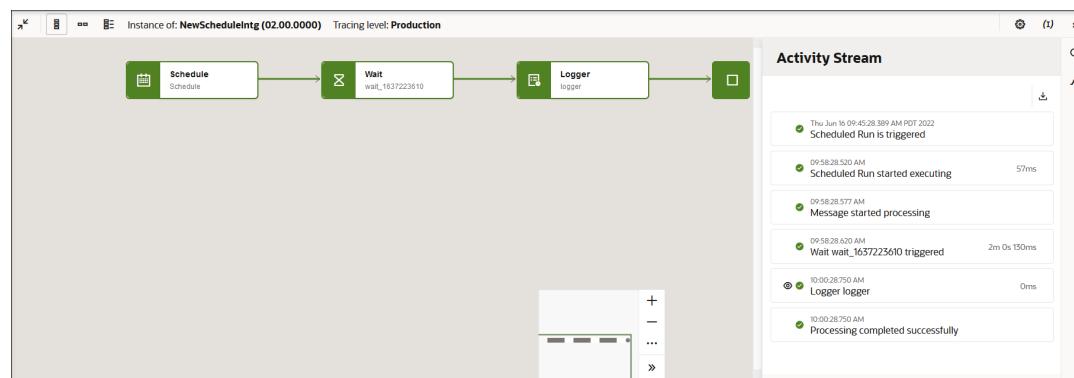
- [View the Instance Elements and Activity Stream on the Instance Details Page](#)
- [View the Results of Iterations of Logic Actions on the Instance Details Page](#)
- [Access Information Through Inline Menus on the Instance Details Page](#)

View the Instance Elements and Activity Stream on the Instance Details Page

- Click a specific business identifier of an integration instance in the **Primary Identifier** column.



The integration flow of the instance and Activity Stream panel are displayed.



- Click **Download**  at the top of the activity stream to download the activity stream logs for the entire instance.
- Click **View Payload**  to view the payload for that part of the integration flow.
- Perform tasks within the payload.

Element	Description
Summary	Click to view a summary of payload details.
Details	Click to view more specific payload details.
Download Payload 	Click to download the payload to your desktop.
Copy Payload 	Click to copy the payload for pasting into a document.

Element	Description
Toggle Line Numbers 	Click to show or hide line numbers in the payload.

5. Click Activity Stream



to close the activity stream.

View the Results of Iterations of Logic Actions on the Instance Details Page

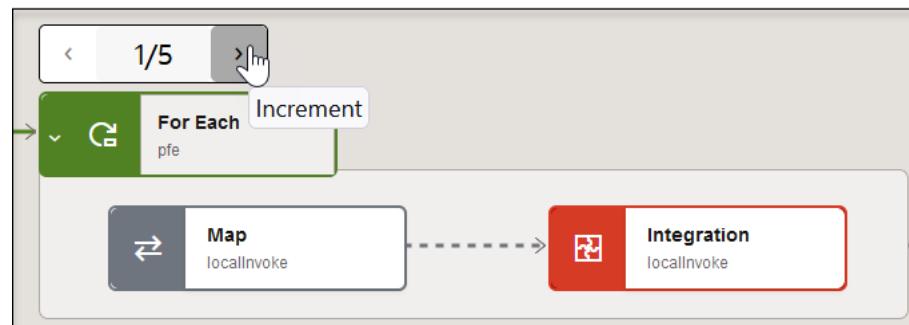
You can view the results of iterations of logic actions (such as a for-each, switch, while, and others) in the activity stream on the Instance Details page.

1. Click a specific business identifier of an integration instance in the **Primary Identifier** column.



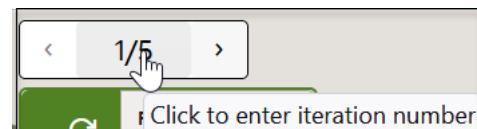
2. Go to the logic action in the integration (for this example, a for-each action is shown).
3. You can move to another iteration of the loop in either of two ways.

- a. Click  to access a view of the next iteration in the loop. For this example, there are five iterations in this for-each action.



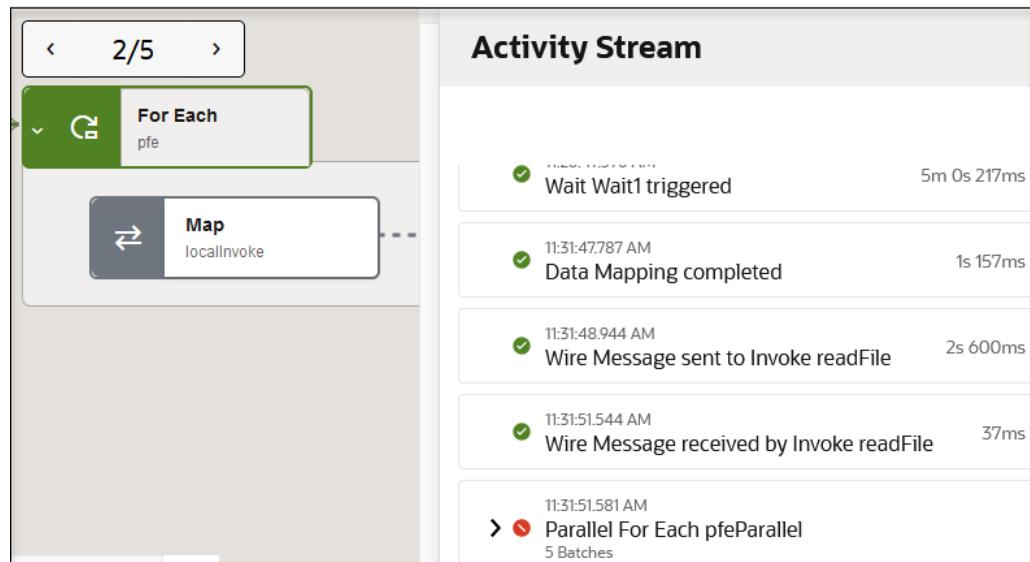
or

- a. Hover over and click the iteration box of a logic action in the integration.



- b. Enter a specific value and click **OK** (for example, 5). This method is useful when the logic action has iterated many times.

The activity stream shows the specified iteration of the logic action.



Access Information Through Inline Menus on the Instance Details Page

The inline menus of the triggers, invokes, and actions in the integration flow provide access to additional information.

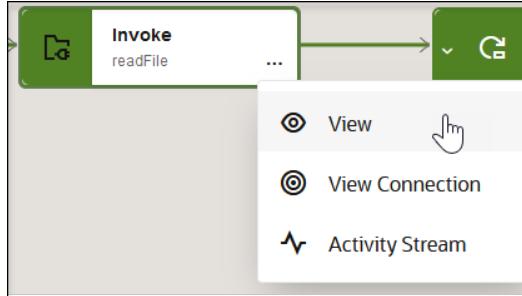
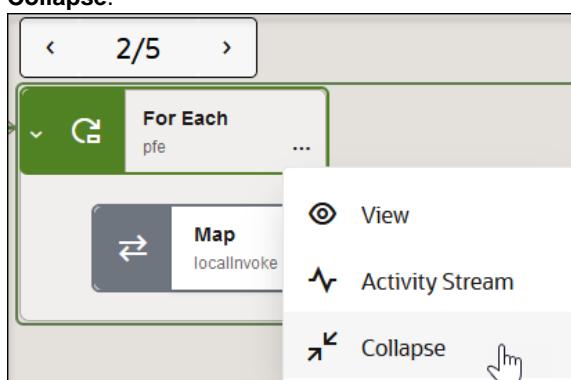
- Click a specific business identifier of an integration instance in the **Primary Identifier** column.

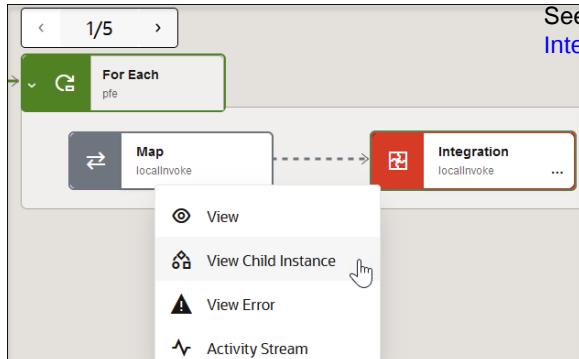


- Select from a set of options. The options that appear are based on the element you selected.

Option	Description
For trigger or invoke connections, click Actions ..., then View Connection .	<p>Displays configuration details for the connection:</p> <ul style="list-style-type: none"> Selected roles for the connection (trigger, invoke, or trigger and invoke)
	<p>Identifier value and date last updated</p> <p>Integrations using the connection (click to view the list)</p> <p>Connection and security properties defined on the Connections page</p> <p>Status of agent group association</p> <p>Configuration progress (the connection properties and security policy must be completely configured and tested to be listed as 100%)</p>

The **Actions** ... menu in the upper right corner enables you to edit or delete the connection (if not currently used in an activated instance).

Option	Description
For trigger, action, and invoke connections, click Actions ... , then View .	<p>Displays the following details:</p> <ul style="list-style-type: none"> For schedules, shows the configured scheduled parameters. For triggers and invokes, the Adapter Endpoint Configuration Wizard is invoked to show the configuration settings. For maps, the mapper is invoked to show the mappings. For other actions, the specific configurations are shown. 
For logic actions (such as a scope, for-each, while, wait, and switch), click Actions ... , then Expand .	Displays each action and invoke within the logic action. 
For logic actions (such as a scope, for-each, while, wait, and switch), click Actions ... , then Collapse .	Displays the collapsed logic action. 

Option	Description
<p>For parent integrations that call child integrations, click Actions ..., then View Child Instances.</p> 	<p>Displays the child integration called by the parent.</p> <p>See Invoke a Child Integration from a Parent Integration.</p>

Determine Why an Integration Instance Stopped Running

When a scheduled or asynchronous integration instance stops running, the reason for the termination appears in the activity stream.

The following sections describe the message displayed in the activity stream based on the type of instance termination.

- [Someone Stopped an Asynchronous Integration Instance](#)
- [Someone Deactivated an Asynchronous Integration While Instances Were Running](#)
- [Someone Stopped a Running Schedule of an Integration](#)
- [Oracle Integration Terminated an Asynchronous or Schedule Instance That Exceeded the Processing Deadline](#)
- [Oracle Integration Terminated a Stuck Scheduled Instance](#)

Someone Stopped an Asynchronous Integration Instance

1. In the navigation pane, click **Observability**, then **Instances**.

2. Hover over the row of the active integration instance to terminate, and select **Abort** .
3. Click **Confirm** when prompted.
4. Click **View Details** .

The Activity Stream panel opens. A red icon and the reason for the termination are displayed.

Instance aborted by username

Someone Deactivated an Asynchronous Integration While Instances Were Running

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the row of the integration instance to deactivate, and select .
3. In the navigation pane, click **Observability**, then **Instances**.
4. Click **View Details** .

The Activity Stream panel opens. A red icon and the reason for the termination are displayed.

Instance aborted due to deactivation of integration by *username*

Someone Stopped a Running Schedule of an Integration

The scheduled run is either currently running successfully or has exceeded six hours in duration (now considered an out-of-bounds run).

1. In the navigation pane, click **Observability**, then **Instances**.
2. Hover over the row of the active, schedule integration instance to terminate, and select .
3. Click **Confirm** when prompted.
4. Click **View Details** .

The Activity Stream panel opens. A red icon and the reason for the termination are displayed.

Instance aborted as schedule was stopped by *username*

Oracle Integration Terminated an Asynchronous or Schedule Instance That Exceeded the Processing Deadline

1. In the navigation pane, click **Observability**, then **Instances**.
 2. In the **Duration** column, find the integration that has exceeded the processing deadline (for example, an integration's duration reached eight hours and two minutes).
 3. Click **View Details** .
- The Activity Stream panel opens and displays the following details for the instance terminated internally by Oracle Integration.

Instance aborted as it has reached maximum duration allow for a flow

Oracle Integration Terminated a Stuck Scheduled Instance

This rare scenario is for scheduler recovery jobs and only applies when a schedule instance is not terminated internally by Oracle Integration.

1. In the navigation pane, click **Observability**, then **Instances**.
 2. In the **Duration** column, find the integration that has exceeded the processing deadline.
 3. Click **View Details** .
- The Activity Stream panel opens and displays the following details for schedule instances.

Instance aborted as it has reached maximum duration allow for a scheduled flow

Replay Integration Instances

You can replay REST Adapter-triggered and AS2 Adapter-triggered integration instances that you configured as replayable during integration activation. This feature enables you to quickly retest your integration without the need to re-enter the data used to initially trigger the integration.

- [Capabilities and Restrictions](#)
- [Understand the Differences Between Replying and Recovering Instances](#)
- [Replay an Integration Instance](#)

Capabilities and Restrictions

- You can only replay instances in which you explicitly select **Allow to run again** in the Activate integration panel.
- You can only replay instances in which the trigger adapter connection is a REST Adapter or AS2 Adapter.
- You can only replay instances with structured payload content. You cannot replay instances with unstructured payload content.
- You can replay instances regardless of their state (for example, you can replay successful, faulted, and aborted instances).
- You can replay instances outside and inside of projects.
- You cannot replay instances where the connectivity agent is used.
- You can only replay a single instance at a time.
- There must be an active integration with the same major version as the replayed instance.
- A parent instance can have many child-replayed instances.
- A replayed instance has a single parent instance.
- You can replay both asynchronous and synchronous instances.
- Payload data for replayed instances is captured for 32 days (for the Production and Audit tracing levels).

Understand the Differences Between Replying and Recovering Instances

Instance replay and instance recovery (performed with the **Resubmit** option on the Errors page) are not the same. Understand the differences to determine which feature is best for your scenario. Instance recovery includes the following restrictions.

- You can only recover a faulted instance that is identified as recoverable.
- You cannot recover synchronous instances.
- You cannot recover scheduled integration instances with scheduled parameters.
- Recovery only updates the existing instance. No new instance is created.
- Recovery is independent of adapter type or payload content.
- Recovery only restarts from the last dehydration point in an instance (for example, a wait action), whereas replay starts completely from the beginning of the instance.

Replay an Integration Instance

This section describes how to replay an integration instance on the Instances page outside of a project. You can also replay an integration instance inside a project. See [Activate or Deactivate an Integration in a Project](#) and [Track Instances in a Project](#).

1. In the navigation pane, click **Design**, then **Integrations**.
2. Hover over the integration to activate, then click **Activate** .
3. Select the level of tracing appropriate to your integration. This level describes the amount of data to capture in the activity stream. See [Activate an Integration](#).
4. In the **Configure runtime options** section, select **Allow to run again**, then click **Activate**. This selection enables you to replay an instance of an integration during runtime on the Instances page. This selection is only available with specific types of integrations. See [Capabilities and Restrictions](#).
5. Run the integration. For REST Adapter trigger connection-based integrations, you can use the Configure and run page. See [Test Integrations from Outside the Integration Canvas](#). The Activity Stream panel opens.
6. Go to the Instances page in either of two ways:
 - a. On the Configure and run page, close the activity stream and click **Track instances**.
 - b. In the navigation pane, click **Observability**, then **Instances**.
7. Hover over the row of the instance to replay, and click **Run again** .

The Run again panel opens.

8. Select the tracing level to use for this instance replay. The level you select is unique to this replay only and does not impact the tracing level you set when activating the integration.
9. Click **Run again**.

When you replay an instance, a new instance ID is created and an icon is added to identify this as a replayed instance. For this example, the replayed instance and the original instance immediately below are shown. You can continue to replay the original instance or the replayed instance.

Primary identifier	Instance ID	Status	Business Identifiers	Duration
cityName: Chicago Get Weather Info 1.0.0	OWdDvI... 	Succeeded 	unit 1 tracking...standard	Received Succeeded Duration Just now Just now 470 m  
cityName: Chicago Get Weather Info 1.0.0	cDTCxF... 	Succeeded 	unit 1 tracking...standard	Received Succeeded Duration 6 minutes ago 6 minutes ago 404 milliseconds

Note:

The **Display instances** option of the **Filter**  enables you to filter on replayed or replayable instances.

10. Click **View details**  to view the activity stream.
11. Click **Actions ...** and note the selections for both the original and replayed instances.
 - The original instance lets you view the replayed child instance.

- The replayed instance lets you view the replayed parent instance and the replayed child instances.

You can replay the original instance again or replay the previously-replayed instance again.

If you no longer want the integration to be replayable, you can change the configuration.

12. In the navigation pane, click **Design**, then **Integrations**.
13. Hover over the row of the integration to configure, and select **Actions ...**, then **Configure activation**.
14. In the **Configure runtime options** section, deselect **Allow to run again**, then click **Save**. You do *not* need to deactivate the integration. The change is applied to the active integration.
15. Run your integration again and go to the Instances page to review the results.
16. Hover over the row for that instance and note that there is no **Run again** icon for replaying this specific instance.

 **Note:**

The instances that you created previously when the integration was replayable still show the **Run again** icon. However, if you attempt to replay these instances, you receive an error. This is because the integration is no longer replayable.

You can also reprocess (replay) B2B for Oracle Integration instances. See B2B Tracking in *Using B2B for Oracle Integration 3*.

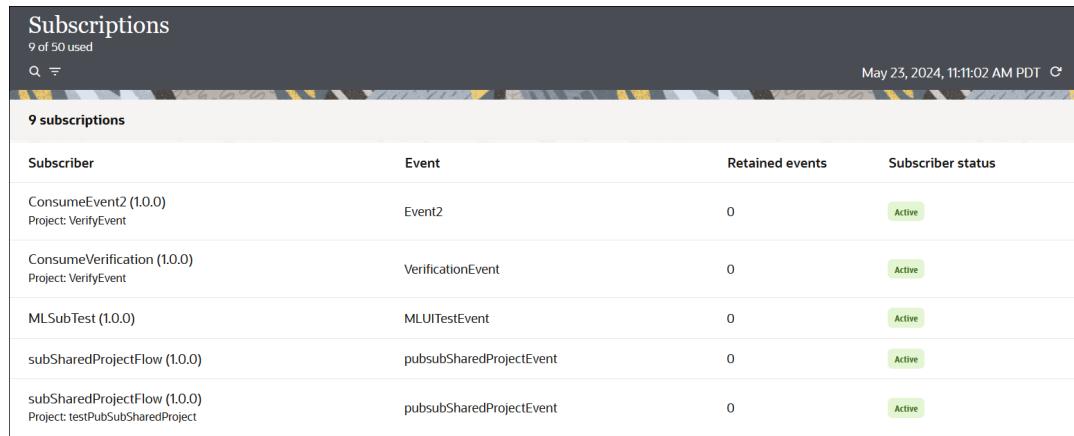
Monitor Integrations That Subscribe to Events

You can monitor the number of integrations that subscribe to events, as well as their status, on the Subscriptions page. This page includes subscribed integrations that are outside of projects (globally available) and inside of projects.

1. In the navigation pane, click **Observability**, then **Subscriptions**.

The Subscriptions page shows the active event subscriptions.

For this example, 9 of the maximum number of 50 integrations allowed for subscribing to events per service instance are currently available. Subscribed integrations inside of projects are designated with the word **Project** in the **Subscriber** column.



The screenshot shows the Subscriptions page with the following details:

Subscriptions			
9 of 50 used	May 23, 2024, 11:11:02 AM PDT	C	
9 subscriptions			
Subscriber	Event	Retained events	Subscriber status
ConsumeEvent2 (1.0.0) Project: VerifyEvent	Event2	0	Active
ConsumeVerification (1.0.0) Project: VerifyEvent	VerificationEvent	0	Active
MLSubTest (1.0.0)	MLUITestEvent	0	Active
subSharedProjectFlow (1.0.0)	pubsubSharedProjectEvent	0	Active
subSharedProjectFlow (1.0.0) Project: testPubSubSharedProject	pubsubSharedProjectEvent	0	Active

- Click **Filter**  to filter by event subscription integrations or retained events (events published, but not yet received by the event subscription integrations).

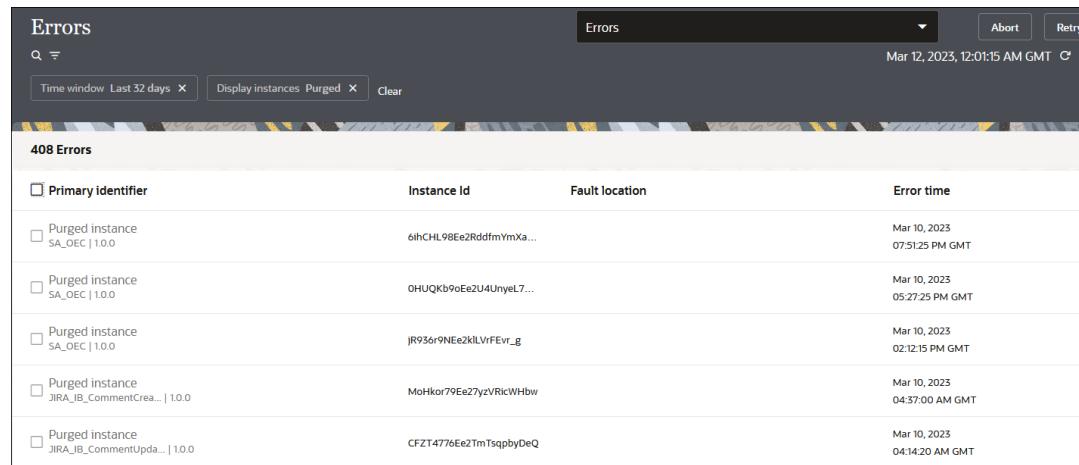
Details about publishing and subscribing to events in integrations are provided. See [Create Event Integrations](#).

View Minimal Details About Debug Tracing Level Instances for 32 Days

When you select the **Debug (not recommended)** option as your tracing level in the Activate integration panel, data is retained for 24 hours before being purged. However, you can view minimal details about the instances for up to 32 days in the Instances page and Errors page. Only the instance ID, instance status, and time duration information are retained. All payload, business identifiers, activity stream, and error message data is purged.

- Go to the Errors page or the Instances page.
 - In the navigation pane, click **Observability**, then **Errors**.
 - In the navigation pane, click **Observability**, then **Instances**.
- From the **Filter** , select **Display instances**, then **Purged**.

On the Errors Page, the instance ID and time of the error are displayed for up to 32 days.



<input type="checkbox"/> Primary identifier	Instance Id	Fault location	Error time
<input type="checkbox"/> Purged instance SA_DEC 1.0.0	6ihCHL98Ee2RddfmYmXa...		Mar 10, 2023 07:51:25 PM GMT
<input type="checkbox"/> Purged instance SA_DEC 1.0.0	0HUQKb9oEe2U4UnyeL7...		Mar 10, 2023 05:27:25 PM GMT
<input type="checkbox"/> Purged instance SA_DEC 1.0.0	jR936r9NEe2kilVrFEvr_g		Mar 10, 2023 02:12:15 PM GMT
<input type="checkbox"/> Purged instance JIRA_IB_CommentCrea... 1.0.0	MoHkor79Ee27yzVRkWHbw		Mar 10, 2023 04:37:00 AM GMT
<input type="checkbox"/> Purged instance JIRA_IB_CommentUpda... 1.0.0	CFZT4776Ee2TmTsqqbyDeQ		Mar 10, 2023 04:14:20 AM GMT

On the Instances page, the instance ID, instance status and instance duration are displayed for up to 32 days.

Instances					Mar 13, 2023, 09:24:08 PM GMT	
<input type="text"/> <input type="button"/>		<input type="button"/> Display instances	<input type="button"/> Purged	<input type="button"/> Clear		
Primary Identifier		Instance Id	Status	Business Identifiers	Received	Duration
Purged instance	AA-B2B-Demo 1.0.0	uTtVYKfG...	Aborted		Feb 8th, 2023 03:39 PM GMT Today at 03:54 PM GMT 10 seconds	
Purged instance	sfdc_23_01_2... sfdc_23_01_2... 1.0.0	Mlx7Obzs...	Aborted		Tuesday at 01:30 PM GMT Today at 03:53 AM GMT 23 min 54 seconds	
Purged instance	sfdc_23_01_2... sfdc_23_01_2... 1.0.0	XWqylME...	Succeeded		Yesterday at 09:03 PM GMT Yesterday at 09:03 PM GMT 939 milliseconds	
Purged instance	sfdc_23_01_2... sfdc_trigger 1.0.0	kj693cEW...	Succeeded		Yesterday at 09:03 PM GMT Yesterday at 09:03 PM GMT 2 seconds	
Purged instance	sfdc_23_01_2... sfdc_23_01_2... 1.0.0	kksaaME...	Succeeded		Yesterday at 08:45 PM GMT Yesterday at 08:45 PM GMT 2 seconds	

Note the following:

- In the **Primary Identifier** column, the **Purged instance** label is displayed for each purged instance.
- You cannot click the purged instance identifier to access graphical details about the instance.
- You cannot perform any actions on this instance, such as resubmitting or aborting the instance on the Errors page, selecting options from **Actions** on the Instances page, and viewing the activity stream on either page. If you select the activity stream, the following message is displayed:

This data is no longer available

View the Calendar of Schedule Integration Runs

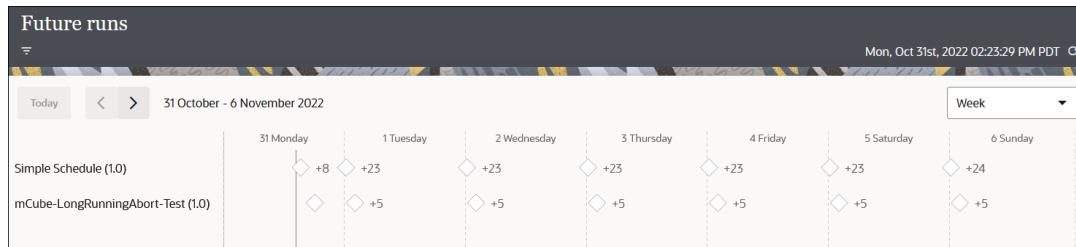
If too many schedule integrations start at the same time, they might experience contention. Ease the contention by staggering the start times. The Future runs page provides an overview of all runs scheduled to trigger in the future, across all schedule integrations whose schedules have started.

 **Note:**

The Future runs page works on the browser time zone, and does not support the selected time zone under the **Preferences** option of the **User** menu in the upper right corner of the page.

- In the navigation pane, click **Observability**, then **Future runs**.

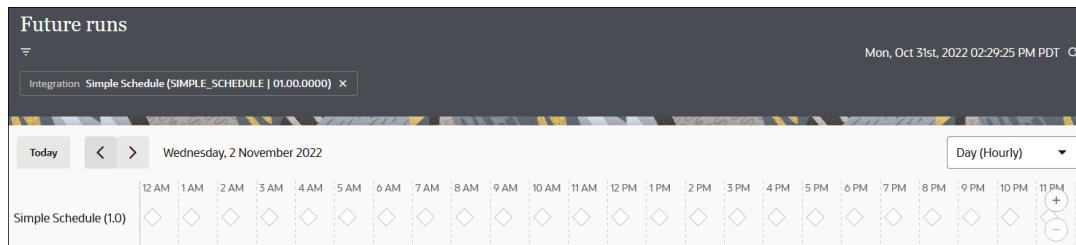
The Future runs page appears.



2. At the far right, use the drop-down list to select the time period to view (by minutes in a day, by hours in a day, by week, or by month).
 - For the week and month views, each icon in the chart represents the first run of that day, followed by the count of runs for that day. For example, +23 means there are 24 runs for that day for a particular schedule. Hovering over the icon shows details, including the time and information about the first run (represented by the icon).
 - For the day view, both by minutes and hours, all future runs are displayed. Every run is represented by the icon. Hovering over them shows the processing time for that run and the sequence (for example, 2 of 20) of that specific run in the day.

You can also zoom in and zoom out, when needed.

3. Click a specific number (for example, **23** for **Simple Schedule** on Wednesday) to show the scheduled runs for the selected time period (for this example, hours in a day).



4. Hover your cursor over each entry to display the schedule integration name, runtime, and run sequence.
5. Click the schedule integration name in the first column to access the Schedule and future runs page for that schedule.
6. Click to filter by integration name and view future runs of that integration.

Diagnose and Manage Event-Based Oracle Fusion Applications Integrations

You can self-diagnose the delivery of business events between Oracle Fusion Applications and Oracle Integration, such as determining if delivery issues are occurring in Oracle Fusion Applications or Oracle Integration. You can also perform some management tasks, such as retrying the delivery of business events that have failed.

The event handling framework (EHF) capability of Oracle Fusion Applications enables application business event propagation beyond the boundary of Oracle Fusion Applications.

When an Oracle Fusion Applications business event is generated, the EHF ensures that the event information is enriched according to the needs of the subscriber and passed to the subscriber across the boundary. In the case of Oracle Integration, the EHF provides business event support for the following adapters configured using the Basic Authentication and OAuth security policies:

- Oracle ERP Cloud Adapter
- Oracle CX Sales and B2B Service Adapter (previously known as the Oracle Engagement Cloud Adapter)

An integration subscribes to business events with these adapters. When the business event is raised in the application, the EHF passes the event payload to Oracle Integration, which invokes the integration.

With the self-diagnostic capabilities of Oracle Integration, you can:

- Diagnose Oracle Fusion Applications event-triggered integrations.
- Minimize the need to file Oracle Fusion Applications escalations and service requests.
- Review and self-service business event integration-related issues.

For example, assume you have the following scenario in which business events are not received by Oracle Integration:

- User A manages logistics for ABC Corporation and is informed about a large shipment dispatch for an important customer.
- User A observes that the notification has not appeared in the application.
- User A contacts User C, who manages Oracle Fusion Applications at ABC Corporation, and shares their concern about the lack of a notification.
- User C sees that the business event has been correctly created.
- However, neither User A nor User C have any idea about how to diagnose where the event disappeared. They consider logging a service request with Oracle for troubleshooting. With the self-diagnostic capabilities in Oracle Integration, these users can self-diagnose message delivery, thereby reducing the need to create a service request for some basic information.

 **Note:**

When business events are not delivered to Oracle Integration, self-diagnosis typically indicates that the error occurred within Oracle Fusion Applications boundaries. Open a service request with Oracle Fusion Applications and include all necessary error messages to aid support in troubleshooting the issue.

1. In the navigation pane, click **Observability**, then **Fusion Applications**.

 **Note:**

The **Fusion Applications** option in the navigation pane is only visible to users with the ServiceAdministrator role. For all other users, this option is not visible.

The Fusion Applications page shows the list of deployed adapter connections and their role (trigger, trigger and invoke). To be shown in the Fusion Application page, the connections must satisfy the following conditions:

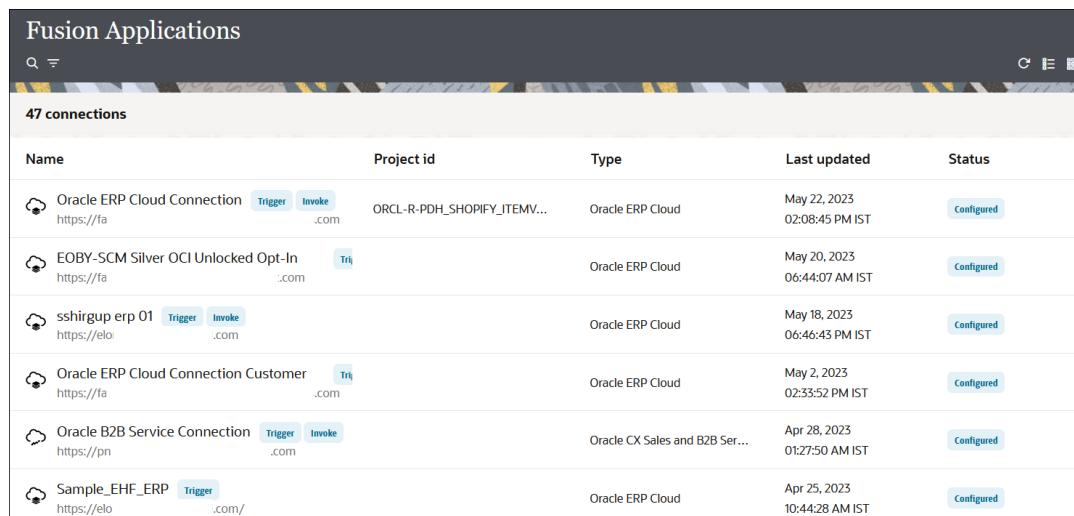
- Must be either the Oracle ERP Cloud Adapter or Oracle CX Sales and B2B Service Adapter.
- Must be a trigger or trigger and invoke role. Invoke alone is not supported.
- Must be in a configured state.

From this page, you can:

- Review basic event information about configured applications.
- Browse the selected application and see subscribed events in detail.
- Review detailed information about failed messages for the selected application.

Key features in the table include the following:

- The **Filter**  enables you to search on connection name, user that created the connection, user that updated the connection, adapter type (Oracle ERP Cloud Adapter or Oracle CX Sales and B2B Service Adapter), and connection role (all, trigger, or trigger and invoke)
- The **Name** column shows the connection name and the Oracle Fusion Applications hostname that was specified in the **Properties** section of the Connections page. This enables you to identify the host on which the event was generated.
- The **Project id** column shows the ID of the project to which a connection belongs. If the connection is not part of any project (known as a connection that is globally available), the column cell is empty.
- The **Type** column shows the adapter used by this connection.



The screenshot shows a table titled "Fusion Applications" with the heading "47 connections". The table has columns: Name, Project id, Type, Last updated, and Status. Each row represents a connection, indicated by a cloud icon. The "Type" column shows various adapter types like Oracle ERP Cloud, Oracle CX Sales and B2B Service, etc. The "Status" column shows "Configured" for most connections. The "Last updated" column provides the date and time of the last update for each connection.

Name	Project id	Type	Last updated	Status
Oracle ERP Cloud Connection https://fa	Trigger Invoke .com	ORCL-R-PDH_SHOPIFY_ITEMV...	Oracle ERP Cloud May 22, 2023 02:08:45 PM IST	Configured
EOBY-SCM Silver OCI Unlocked Opt-In https://fa	Trigger .com	Oracle ERP Cloud May 20, 2023 06:44:07 AM IST	Configured	
sshirgup erp 01 https://elo	Trigger Invoke .com	Oracle ERP Cloud May 18, 2023 06:46:43 PM IST	Configured	
Oracle ERP Cloud Connection Customer https://fa	Trigger .com	Oracle ERP Cloud May 2, 2023 02:33:52 PM IST	Configured	
Oracle B2B Service Connection https://pn	Trigger Invoke .com	Oracle CX Sales and B2B Ser... Apr 28, 2023 01:27:50 AM IST	Configured	
Sample_EHF_ERP https://elo	Trigger .com/	Oracle ERP Cloud Apr 25, 2023 10:44:28 AM IST	Configured	

2. Hover over a connection row, and click **View event list for the Fusion Applications instance**  to view details about business event subscriptions on the Oracle Fusion Applications host. See [Monitor Business Event Delivery Status](#).

Monitor Business Event Delivery Status

You can view details about business event subscriptions, including their status (queued, delivered, retried, and retries that have reached their limit) and potential reasons for why a

business event has not been delivered to Oracle Integration. You can also retry the delivery of business events.

The screenshot shows the Oracle Fusion Applications Status overview page. At the top, it says 'Events for https://fa- .com' and 'Includes events for Sample_EHF_ERP'. Below that is a 'Status overview' section with a 'Time window Last 7 days' button. The main area shows '1 events' with a table:

Name	Subscriptions	Queued	Delivered	Retried	Maxed out retries
Account Created Event	5 integrations	0	15	0	2

1. Click **Filter** to adjust the time range for gathering event details.
2. Hover your cursor over a business event row, and click **View** .
3. Click **Filter** if you want to change the time range and the status of business events to display:
 - **Maxed out retries** (the default status)
 - **Retried**
 - **Queued**
 - **Delivered**
4. Click **Apply**.

For the following example, **Maxed out retries** is selected and business events that have reached their retry limit are displayed.

The screenshot shows the 'Account Created Event' details page. It has a 'Event details' header with a 'Time window Last 7 days' button and a 'Status Maxed out retries' button. Below that is a '2 event instances' section with a table:

Event instance id	Subscriber	Date	Reason
3248685 CreateOrganization	N/A (1.0)	Mar 28, 2023 11:05:28 AM IST	Unable to invoke endpoint URI "https://fs
3248685 CreateOrganization	N/A (1.0)	Mar 28, 2023 11:05:28 AM IST	Unable to invoke endpoint URI "https://fs

5. Click **Open Details**

A reason for the error is displayed. For this example, the business event never triggered the integration in Oracle Integration. This may be because the message is stuck in Oracle Fusion Applications or there is a networking issue.

Account Created Event

Event details

Time window Last 7 days Status Maxed out retries

2 event instances

Event instance id	Subscriber	Date	Reason
3248685 CreateOrganization	N/A (1.0)	Mar 28, 2023 11:05:28 AM IST	Unable to invoke endpoint URI "https://.com/ic/ws/integration/v2/flows/osc/ACCOUNTOEC_TO_CUSTOMERNE/1.0/" successfully due to: javax.xml.soap.SOAPException: Bad response: 404 Not Found from url https://.com/ic/ws/integration/v2/flows/osc/ACCOUNTOEC_TO_CUSTOMERNE/1.0/

REASON

Unable to invoke endpoint URI "https://.com/ic/ws/integration/v2/flows/osc/ACCOUNTOEC_TO_CUSTOMERNE/1.0/" successfully due to: javax.xml.soap.SOAPException: Bad response: 404 Not Found from url https://.com/ic/ws/integration/v2/flows/osc/ACCOUNTOEC_TO_CUSTOMERNE/1.0/

- If you want to attempt business message delivery again, select the event instances to retry and click **Retry**. You can also click **Event instance ID** to retry all event instances. The maximum number of retries is 10. After that, the event's status is moved to **Maxed out retries**. The event then requires a manual intervention (retry). Once you click **Retry**, the event is queued for retry, but its state has not yet changed. Therefore, that event is still shown. You must refresh the page and check to see if the event's status has changed.

Send System Status Reports with Notification Emails

You can notify users by email with hourly or daily reports about total messages received, total messages processed, successful messages, failed messages, and successful message rate; with service failure alerts (for example, and when a service is down or an agent becomes unavailable). You can also remove email addresses from the suppression list.

 **Note:**

This is a system level setting. Therefore, user A cannot create one notification setup while user B creates another notification setup. User B can see the notification setup created initially by user A and can modify it.

Topics:

- Understand DKIM Authentication and SPF Record Details
- Configure Notification Emails
- Remove Email Addresses from the Suppression List
- Change the Delivery Time of Notification Emails

Understand DKIM Authentication and SPF Record Details

When you configure notification emails to send service failure alerts, system status reports, and integration error reports, you can optionally specify Domain Keys Identified Mail (DKIM) authentication and Sender Policy Framework (SPF) record details.

Details and examples are provided to better understand DKIM and SPF. See the [An Advanced Guide to OIC Notification via Emails](#) blog and Configure Email Authentication Settings for SPF and DKIM in *Provisioning and Administering Oracle Integration 3*.

Configure Notification Emails

You can configure notification emails to notify interested parties about system status on a regular basis and alert them about service failures. Detailed system status reports include metrics on instances and runs processed by the system, including failures. You can select to send an hourly or daily report and specify the list of recipients.

The report includes the following information:

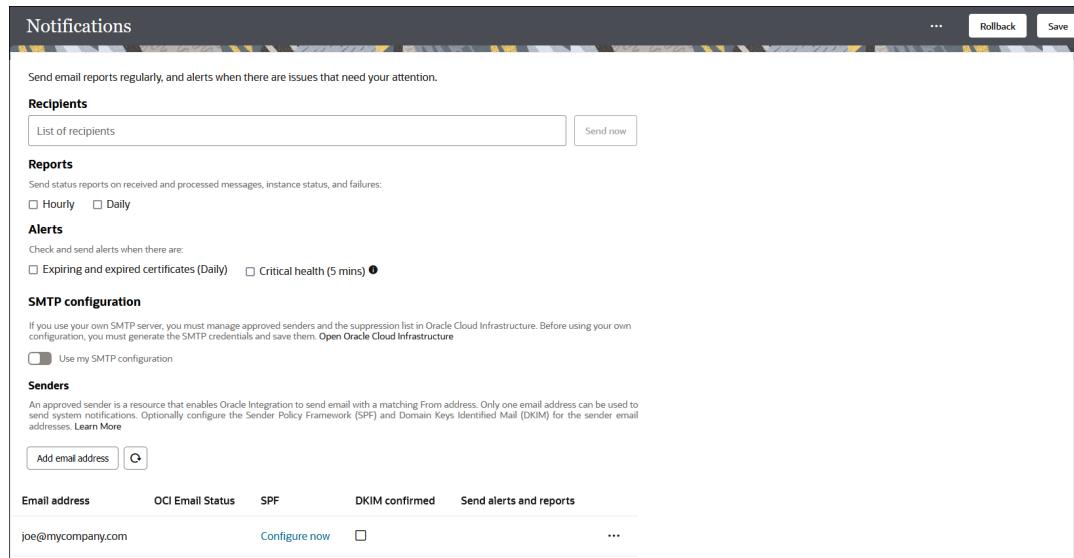
- POD name (also included in the subject of the email)
- Time period
- Total messages received, processed, succeeded, aborted, and failed
- Percentage of successful messages
- Errors by integration instance and run including messages received, processed, succeeded, and failed

The email includes a link at the bottom to change the notification settings or completely stop receiving email notifications. Click **Oracle Integration** to access the Notifications page to adjust your settings.

The Notifications page enables you to manage email delivery through either of two methods:

- The default method: Sets your limit at 10,000 emails for a rolling 24-hour window. This is provided automatically and requires no additional configuration to use.
- Your own customer tenancy: Sets your limit to the number allowed by the Oracle Cloud Infrastructure Email Delivery Service for a rolling 24-hour window. See [Email Delivery Service Capabilities and Limits](#). To use your customer tenancy, you must do the following:
 - Create SMTP credentials in the Oracle Cloud Infrastructure Console in order to complete the **Use customer SMTP configuration section**. Only Oracle Cloud Infrastructure SMTP credentials are supported. See [Generate SMTP Credentials for a User](#).
 - Register the default sender email address and the sender email address for system notification with Oracle Cloud Infrastructure in your customer tenancy.
 - Manage the approved senders and suppression list in your customer tenancy.
 - Check the service limits. See [Overview of the Email Delivery Service](#).

1. In the navigation pane, click **Settings**, then **Notifications**.



2. In the **Recipients** field, enter the email address to which to send the reports. You can specify multiple email addresses, each separated by commas.
3. In the **Reports** field, select the type of report to send.
 - **Hourly:** Send a detailed report every hour.
 - **Daily:** Alerts are sent once a day.
4. In the **Alerts** section, select to send alerts when the following occurs:

Option	Description
Expiring and expired certificates (Daily)	<p>Select to send alerts when the following occurs:</p> <ul style="list-style-type: none"> • Certificates are set to expire within seven days. • Certificates have expired. <p>The check is made for certificates you uploaded on the Certificates page. Certificate expiration status is checked daily. Oracle Integration provided-certificates are not checked.</p>
Critical health (5 mins)	<p>Select to send alerts when one or more of the following conditions are detected:</p> <ul style="list-style-type: none"> • The email quota exceeded the threshold. • One or more connectivity agent instances became unavailable.

5. To use your own customer tenancy to manage the number of emails to send, click to enable customer tenancy mode.
 - a. Specify the SMTP user name and password and the default sender email address (that is, the `from` address). These three fields are required. You can also optionally specify a default address for system notifications.

SMTP configuration

If you use your own SMTP server, you must manage approved senders and the suppression list in Oracle Cloud Infrastructure. Before using your own configuration, you must generate the SMTP credentials and save them. [Open Oracle Cloud Infrastructure](#)

Use my SMTP configuration Test

Configure SMTP details

Prior to saving, test your configuration which validates it by sending an email to the specified recipient(s).

SMTP username	Required	
SMTP password	<input type="password"/> (eye)	Required
Default sender email address	Required	
Sender email address for system notification	Required	

- b. Click **Test** to validate the SMTP credentials.
- c. Click **Save**.

 **Note:**

If you want to return to using the default method, click to disable the SMTP credentials, and click **Save**.

6. To use the default method to manage the number of emails to send, follow these steps:
 - a. In the **Senders** section, click **Add email address** to add approved senders.
 - b. Specify the following details:

Element	Description
Email address	An approved sender enables Oracle Integration to send email with a matching From address. An email is sent to each of the email addresses listed as approved senders. Upon approval, these email addresses can be used as the From address in a notification action of an integration and/or system notification.

Element	Description
OCI Email Status	<p>Perform the following steps:</p> <ul style="list-style-type: none"> i. Enter an approved sender email address and click Save. A verification code is sent to that address and verification status changes to In Progress. ii. Click Enter verification code and enter the code received by that email address. Upon successful verification, status is changed to Verified. <p>Note the following:</p> <ul style="list-style-type: none"> • Existing sender emails that were previously approved, but never formally verified with this process, can continue to be used for now. It is recommended that you delete these entries, re-enter them, and enter the verification code sent to the email address. • Don't share approved sender email addresses across service instances (for example, test and production instances). Create different approval emails addresses for each service instance, even if they are in the same tenancy.
SPF	Optionally configure the Sender Policy Framework (SPF) for the sender email addresses. You configure SPF support outside of Oracle Integration.
DKIM confirmed	Optionally confirm the Domain Keys Identified Mail (DKIM) for the sender email addresses. You configure DKIM support outside of Oracle Integration.
Send alerts and reports	Select to use this email address for the system notifications you selected at the top of the page. However, only a single email address can be selected to send system notifications.

Recommendations are provided to ensure that you configure notification email correctly. See [Troubleshoot Oracle Cloud Infrastructure Notification Email Configuration to Ensure Proper Delivery](#).

- c. Click **Save**.
7. Perform other email notification tasks on this page.

To ...	Description
Receive an immediate email about system status	Click Actions • • • and select Send Now .
Clear the distribution list, selected alert/reports options, and specified sender email addresses.	Click Actions • • • and select Reset All Notifications .
Remove email addresses from the suppression list Note: This task is only required if you are using the default method.	Click Actions • • • and select Email Suppression List . See Remove Email Addresses from the Suppression List .

Remove Email Addresses from the Suppression List

You can remove email addresses from the suppression list. The suppression list lets you refresh each individual email address to see the current state. You can also see the date on which an email sent to that email address was last suppressed. The date is refreshed each time an email is sent to that address.

Adding users to a suppression list is a common task in email address management. Reasons for adding email addresses to the suppression list can include the following:

- An employee on an email distribution list left the company and their email address was deactivated.
- SPF and DKIM not being configured on the sender's address domain server (DNS) result in soft bounces and subsequently end up on the suppression list.

Oracle Integration provides a self-service feature that enables you to view and remove email addresses from the suppression list, when necessary. This feature eliminates the need to file a service request to remove addresses from the suppression list.

Suppressed email addresses can lead to errors in your integration. The activity stream of the instance can show that a notification action failed if the email address used to send the notification is on the suppression list.

1. Click Actions . . . and select Email suppression list.

The Email suppression list panel opens to show all email addresses that have been suppressed. Email addresses on the suppression list cannot send or receive email notifications.

2. Browse or search for the email address to remove.

3. Click Refresh  to ensure that you are seeing the latest state for this specific email address in this service instance. Performing a refresh may remove that email address (because the suppression list was updated behind the scenes), change the email address state message (for example, from hard bounces to repeated soft bounces), or update the timestamp.

4. Next to the email address to remove, click Delete .

This action removes this address from the suppression list. You *cannot* remove more than one address at a time.

Change the Delivery Time of Notification Emails

You can change the delivery time of notification emails.

The tracking metrics daily report runs by default at 00:00 UTC time, which can differ from your time zone. For example, the central time zone is 6 hours behind the UTC time zone. When the report runs at 00:00 UTC on November 14th, it is processing at 6:00pm central time on November 13th.

The report time is based on UTC time, which ensures a common reference point (particularly for the log files). This enables anyone in any time zone to identify the time of processing. This is important for global companies with operations in multiple countries/time zones.

As a workaround, set up notifications to run hourly and use a tool such as an Outlook rule to automatically delete the 23 unwanted emails, keeping the one email with the tracking metrics you want.

See [Troubleshoot Oracle Cloud Infrastructure Notification Email Configuration to Ensure Proper Delivery.](#)

13

Troubleshoot Oracle Integration

Learn about Oracle Integration troubleshooting tasks, including configuration recommendations to ensure proper email delivery, connectivity agent issues, integration runtime issues, integration design-time issues, schedule integration issues, and integration activation issues.

Notes:

- There are overall service limits for Oracle Integration. A service limit is the quota or allowance set on a resource. See *Service Limits in Provisioning and Administering Oracle Integration 3*.
- [Cloud Customer Connect](#) provides a community in which to ask questions, connect with experts, and share your thoughts and ideas about Oracle Integration.

Topics:

- [Troubleshoot Oracle Cloud Infrastructure Notification Email Configuration to Ensure Proper Delivery](#)
- [Troubleshoot the Connectivity Agent](#)
- [Troubleshoot Private Endpoints](#)
- [Troubleshoot Integration Runtime](#)
- [Troubleshoot Integration Design Time](#)
- [Troubleshoot Schedule Integrations](#)
- [Troubleshoot Integration Activations](#)

[Troubleshoot Oracle Cloud Infrastructure Notification Email Configuration to Ensure Proper Delivery](#)

Follow these recommendations to correctly configure and use the default `from` address and suppression list. These recommendations help you to avoid email delivery issues.

- [Default From Address](#)
- [Suppression List](#)
- [Invalid Email Addresses in Distribution Lists \(DL\)](#)

Default From Address

- Don't use `no-reply@oracle.com` as the `from` address.
- Don't use the oracle domain.

- Change the default from address from no-reply@oracle.com to no-reply@mail.integration.region.ocp.oraclecloud.com. The region attribute is provided by Oracle Integration.
- Change the from address in your integrations from no-reply@oracle.com to no-reply@mail.integration.region.ocp.oraclecloud.com. The region attribute is provided by Oracle Integration.

Suppression List

- Add To addresses to the suppression list for a number of reasons:
 - As of now, the recipient address when a hard bounce occurs (emails go undelivered for permanent reasons), when a soft bounce occurs (emails go undelivered for temporary reasons), and when a large number of emails are received are some of the reasons to add the To address to the suppression list.
- If DomainKeys Identified Mail (DKIM) and Sender Policy Framework (SPF) are not configured for the from address domain, the likelihood of having a bounce or messages being silently dropped by the receiving infrastructure is higher.
- You can remove email addresses from the suppression list. See [Remove Email Addresses from the Suppression List](#).

Invalid Email Addresses in Distribution Lists (DL)

If an email DL includes an invalid email address, *all* email addresses in the list are impacted. When a notification alert is sent, none of the recipients in the DL receive the alert.

Troubleshoot the Connectivity Agent

This section describes how to troubleshoot and resolve connectivity agent issues in Oracle Integration.

Topics:

- [Troubleshoot Agent Unavailable Issues](#)
- [Unable to Start Connectivity Agent Because Agent Status Cannot Be Obtained](#)
- [Connectivity Agent Startup Fails With a "java.lang.RuntimeException: Agent Startup Failed - Unable to get agent status" Error](#)
- [Connectivity Agent Installation Error if /etc/hosts File is Not Configured](#)
- [Restrictions on Using the Stage File Action with the Connectivity Agent](#)
- [Java Memory Errors](#)
- [Add or Change the Non-Proxy Host Configuration After Installation in the CpiAgent.properties File](#)
- [Class Loading Conflict When Sharing the JDK Instance with Another Product](#)
- [Connectivity Agent Log File Location](#)
- [Agent Behavior in a Decommissioned Instance or HTTP 404/401 Error Response Codes](#)
- [Verify Endpoint Accessibility when Agent is Installed with a Proxy Host](#)
- [Unlock the Agent Group](#)
- [Failure to Send a Response Due to java.net.SocketException: Connection Reset Error](#)

- Agent Installation on Linux Fails When Using an Installer Copied Using winscp
- Integration Activation Error Due to Change in JDK Location
- Troubleshoot Network Connectivity Issues
- HTTP 401 Unauthorized Error Occurs During Connectivity Agent Installation

Troubleshoot Agent Unavailable Issues

Connectivity agent status can be determined by selecting **Observability > Agents** to access the Monitor agents page. If the agent status is displayed as unavailable, follow this troubleshooting workflow.

Number	Step	Details
1	Check the agent process is running	<p>Check whether the agent is running. Run one of these commands from the host where the agent is installed.</p> <p>If you don't see <code>connectivityagent.jar</code> as a listed process, the agent is not running. For example: 19054 <code>connectivityagent.jar</code></p> <p>Linux:</p> <pre>ps -ef grep connectivityagent.jar</pre> <p>Windows:</p> <pre>jps -l</pre> <p>If the agent process is not running, start it:</p> <p>Linux:</p> <p>Start the connectivity agent as a background process:</p> <pre>nohup java -jar connectivityagent.jar &</pre> <p>Windows:</p> <pre>java -jar connectivityagent.jar</pre>

 **Note:**

Starting the agent as a Windows service is not supported.

If you cannot start the agent, continue to check network connectivity.

If the agent process is running, restart it:
Follow these steps to restart the agent.

Number	Step	Details
2	Check network connectivity	<p>Check that the agent host can connect to the Oracle Integration instance:</p> <p>On the host where the agent is installed, run this command, replacing with your own path:</p> <pre>curl https://myoic-tenancy-ia.integration.ocp.oraclecloud.com/ic/home</pre> <p>If the response is 302 Found, the agent is able to resolve the Oracle Integration instance host. You have network connectivity.</p> <p>If the response is not 302 Found, there may be network issues. Check the /etc/hosts file:</p> <ol style="list-style-type: none"> 1. Check for the correct IP address and host name for your Oracle Integration instance. 2. Troubleshoot with your network administrator to check that connectivity to the instance is not blocked by a firewall or proxy. Here's agent troubleshooting information.
3	Download the Configuration File	Download the agent configuration file again and restart the connectivity agent after replacing the previous agent configuration file.
4	Log a Service Request	<p>If you've gone through the troubleshooting steps, can't find anything in the diagnostic log, and the agent status is still unavailable, create a service request with Oracle Support.</p> <p>Include the diagnostic log in the service request.</p>

Unable to Start Connectivity Agent Because Agent Status Cannot Be Obtained

Connectivity agent startup can fail with the following error:

```
java -jar connectivityagent.jar &

[2024-12-09T19:30:16.392Z] [INFO] [ThreadID: 15] [Logger: oracle.wsm.agent.WSMAgent] [SRC Class: oracle.wsm.agent.WSMAgent; Method: processRequest] pool-1-thread-1Entering processRequest() - 12/09/2024 19:30:16:392
[2024-12-09T19:30:16.396Z] [INFO] [ThreadID: 15] [Logger: oracle.wsm.agent.WSMAgent] [SRC Class: oracle.wsm.agent.WSMAgent; Method: processRequest] pool-1-thread-1Exiting processRequest() - 12/09/2024 19:30:16:396
[2024-12-09T19:30:16.473Z] [INFO] [ThreadID: 15] [Logger: oracle.cloud.cpi.agent.util.CpiUtils] [SRC Class: oracle.cloud.cpi.agent.util.CpiUtils; Method: createAuthToken] OAuth token url https://idcs-xxxxxx.identity.oraclecloud.com:443/oauth2/v1/token
[2024-12-09T19:30:46.505Z] [SEVERE] [ThreadID: 15] [Logger: oracle.cloud.cpi.agent.util.CpiUtils] [SRC Class: oracle.cloud.cpi.agent.util.CpiUtils; Method: createAuthToken] IDCS Token Generation: Call to create token failed due to java.net.SocketTimeoutException: Connect timed out
[2024-12-09T19:30:46.507Z] [INFO] [ThreadID: 15] [Logger: oracle.wsm.agent.WSMAgent] [SRC Class: oracle.wsm.agent.WSMAgent; Method:
```

```

processRequest] pool-1-thread-1Entering processRequest() - 12/09/2024
19:30:46:507
[2024-12-09T19:30:46.507Z] [INFO] [ThreadID: 15] [Logger:
oracle.wsm.agent.WSMAgent] [SRC Class: oracle.wsm.agent.WSMAgent; Method:
processRequest] pool-1-thread-1Exiting processRequest() - 12/09/2024
19:30:46:507
[2024-12-09T19:30:46.550Z] [SEVERE] [ThreadID: 15] [Logger:
oracle.cloud.cpi.agent.install.AgentInstaller] [SRC Class:
oracle.cloud.cpi.agent.install.AgentInstaller; Method: checkVersion] Agent
Installation Failed !!! Incorrect OIC Username / Password provided.
[2024-12-09T19:30:46.551Z] [SEVERE] [ThreadID: 15] [Logger:
oracle.cloud.cpi.agent.ProxyCpiAgentLoader] [SRC Class:
oracle.cloud.cpi.agent.ProxyCpiAgentLoader; Method: configureAgent] Agent
Installation Failed !!! java.lang.reflect.InvocationTargetException
at java.base/
jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at java.base/jdk.internal.reflect.NativeMethod

```

This error occurs because the connectivity agent cannot query the agent status from the Oracle Integration server.

1. Allowlist your identity domain or Oracle Identity Cloud Service, design time IP address, and runtime IP address.
2. Check connectivity to Oracle Integration 3.
3. (Optional) Download the configuration file again and restart the agent if no issue is found with Oracle Integration 3 connectivity.

Connectivity Agent Startup Fails With a "java.lang.RuntimeException: Agent Startup Failed - Unable to get agent status" Error

Connectivity agent startup can fail with the following error:

```

[2021-06-19T10:06:06.567Z] [SEVERE] [ThreadID: 1] [Logger:
oracle.cloud.cpi.agent.CpiBootstrapLoader]
[SRC Class: oracle.cloud.cpi.agent.CpiBootstrapLoader; Method: main] Error
encountered !!! java.lang.RuntimeException:
java.lang.reflect.InvocationTargetException
at
oracle.cloud.cpi.agent.AgentStartCommandTask.run(CpiBootstrapLoader.java:452)
at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511)
at java.util.concurrent.FutureTask.run(FutureTask.java:266)
at
java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.access$20
1(ScheduledThreadPoolExecutor.java:180)
at
java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.run(Sched
uledThreadPoolExecutor.java:293)
at
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149
)
at
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624
)
at java.lang.Thread.run(Thread.java:748)

```

```
Caused by: java.lang.reflect.InvocationTargetException
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.j
ava:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at
oracle.cloud.cpi.agent.CpiBootstrapLoader.startAgent(CpiBootstrapLoader.java:1
92)
at
oracle.cloud.cpi.agent.AgentStartCommandTask.run(CpiBootstrapLoader.java:436)
...
...
Caused by: java.lang.reflect.InvocationTargetException
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.j
ava:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at oracle.cloud.cpi.agent.CpiAgentLoader.startAgent(CpiAgentLoader.java:27)
...
...
Caused by: java.lang.RuntimeException: Agent Startup Failed - Unable to get
agent status :
at
oracle.cloud.cpi.agent.ProxyCpiAgentLoader.initialize(ProxyCpiAgentLoader.java
:151)
at
oracle.cloud.cpi.agent.ProxyCpiAgentLoader.runAgent(ProxyCpiAgentLoader.java:5
3)
...
...
```

Ensure that connectivity from your agent host to Oracle Integration 3 is working.

1. Allowlist your identity domain or Oracle Identity Cloud Service, design time IP address, and runtime IP address.
2. Check connectivity to Oracle Integration 3.
3. (Optional) Download the configuration file again and restart the agent if no issue is found with Oracle Integration 3 connectivity.

Connectivity Agent Installation Error if /etc/hosts File is Not Configured

During connectivity agent installation, if the agent installer cannot find details about the host name/virtual machine (VM) on which the agent is being installed, you may receive the following error:

```
Caused by: java.net.UnknownHostException: : Unknown
name or service
```

Resolve this error by adding an entry for the IP address and host name in the `/etc/hosts` file of the host name/VM.

Restrictions on Using the Stage File Action with the Connectivity Agent

Be aware of several restrictions when using some operations of the stage file action with the connectivity agent. See [Restrictions on Using Stage File Action Operations with the File/Attachment Features of the Connectivity Agent](#).

Java Memory Errors

The agent Java Virtual Machine can stop running when allocating memory with the following seemingly out-of-memory error:

```
There is insufficient memory for the Java Runtime Environment to continue.  
-----  
Java HotSpot(TM) 64-Bit Server VM warning: INFO:  
os::commit_memory(0x00007f6847af000, 12288, 0) failed; error='Cannot  
allocate memory' (errno=12)  
#  
# There is insufficient memory for the Java Runtime Environment to continue.  
# Native memory allocation (mmap) failed to map 12288 bytes for committing  
reserved memory.  
# An error report file with more information is saved as:
```

However, this issue is unrelated to the out-of-memory error that is sometimes observed when the Java heap is not large enough.

This error occurs when Java requests more memory from the operating system, which does not have any:

```
# There is insufficient memory for the Java Runtime Environment to continue.  
# Native memory allocation (mmap) failed to map
```

This may be related to resources on the physical machine/virtual machine/server on which the agent is running. As an example, when the agent is running on the same virtual machine as the database, and the database may be consuming most of the resources. Oracle recommends that you set up the agent on a separate compute.

Add or Change the Non-Proxy Host Configuration After Installation in the CpiAgent.properties File

If you need to add, change, or bypass the non-proxy host configuration *after* agent installation, do *not* edit the `InstallerConfig.cfg` file. Proxy host changes made to that file after agent installation do not take effect. Instead, update the host with the `proxy_nonProxyHosts` parameter in the `Agent_Installation_Location/agenthome/agent/config/CpiAgent.properties` file for your changes to take effect. After editing this file, restart the agent.

Class Loading Conflict When Sharing the JDK Instance with Another Product

When using the connectivity agent, the following error occurs because your JDK instance is shared with another product for which JAR files have been added in the JDK's endorsed directory. This results in a class loading conflict with the agent:

```
ClassCastException: com.sun.xml.messaging.saaj.soap.ver1_1.Message1_1Impl  
cannot be cast to oracle.j2ee.ws.saaj.soap.MessageImpl
```

Ensure that your connectivity agent is running with a JDK installation that is not modified because of use with other products.

Connectivity Agent Log File Location

The connectivity agent `agent-diagnostic0.log` file is available under `agenthome/logs`.

Agent Behavior in a Decommissioned Instance or HTTP 404/401 Error Response Codes

The following code in the logs indicates that agent runtime message processing has halted. This occurs if an HTTP 404/401 error code is continuously received by the agent for 24 hours. The decommission of an Oracle Integration instance also triggers this behavior. When the conditions leading to this error have been resolved, the agent must be restarted manually. See [Restart the Connectivity Agent](#).

```
[2018-10-13T04:30:13.501Z] [SEVERE] [ThreadID: 18] [Logger:  
oracle.cloud.cpi.agent.transport.AQRuntimeConsumer] [SRC Class:  
oracle.cloud.cpi.agent.transport.AQRuntimeConsumer; Method: run] Terminate  
flag activated. Signalling termination of agent runtime poller thread with Id  
  
CPI_RUNTIME_REQ_RES_CHANNEL_worker_0.
```

Verify Endpoint Accessibility when Agent is Installed with a Proxy Host

When the agent is installed with a proxy host, carefully check that the endpoint to access through the agent is reachable through the proxy host. If it is not reachable through the proxy host, you must configure the on-premises endpoint host in the `proxy_NON_PROXY_HOSTS` parameter of the `Agent_Installation_Location/agenthome/agent/config / CpiAgent.properties` file.

Unlock the Agent Group

When an agent group is in edit mode and the browser crashes, the agent group becomes locked, which prevents it from being edited. This results in the following error:

```
ICS-10507: The agent group cannot be updated because it is locked.
```

To unlock the agent group:

1. Log in again as the same user who was editing the agent group when the browser crashed, then log out. This action unlocks the agent group.
or
1. Wait 30 minutes for the lock to expire after the timeout starts.

Failure to Send a Response Due to a java.net.SocketException: Connection Reset Error

For connectivity agent installations running on an Oracle Integration Classic VM and connecting to Oracle Integration (running on Oracle Cloud Infrastructure), the design-time and runtime operation involving the connectivity agent sometimes fails with a `java.net.SocketException: Connection reset`.

This can occur because of a Maximum Transmission Unit (MTU) mismatch.

Here is the complete error:

```
[2019-01-03T16:35:12.670Z] [SEVERE] [ThreadID: 50] [Logger:  
oracle.cloud.cpi.agent.transport.CpiAgentAQTransporter] [SRC Class:  
oracle.cloud.cpi.agent.transport.CpiAgentAQTransporter; Method:  
sendOneWayPacket] Exception while sending response back to  
ICSjava.net.SocketException: Connection reset  
com.sun.jersey.api.client.ClientHandlerException: java.net.SocketException:  
Connection reset  
at  
com.sun.jersey.client.urlconnection.URLConnectionClientHandler.handle(URLConne  
ctionClientHandler.java:155)  
at com.sun.jersey.api.client.Client.handle(Client.java:652)  
at com.sun.jersey.api.client.WebResource.handle(WebResource.java:682)  
at com.sun.jersey.api.client.WebResource.access$200 (WebResource.java:74)
```

To make this connection work, set the MTU of the Oracle Integration Classic VM (where the agent is installed) to 1500 from the current value of 8900.

Perform the following steps:

1. Run `ifconfig -a` as the root user (sudo) and note down the network interface.
2. Run the following command as the root user (sudo) for the network interface (assuming the network interface is `eth0`):

```
ifconfig eth0 mtu 1500
```

Anytime the agent VM is restarted (note that it is not an agent restart, but the host where the agent is installed), the changes must be done for the network interface before restarting the agent.

Agent Installation on Linux Fails When Using an Installer Copied Using winscp

During connectivity agent installation in Linux environments, the installation sometimes fails with the following error:

On premise agent is throwing the following error: `java.lang.RuntimeException: Agent Startup Failed - java.lang.IllegalArgumentException: URI is not absolute`

Installation failure occurs for the following reason:

1. The agent is to be installed in a Linux environment.

2. The agent installer ZIP file is downloaded in a Windows environment and transferred to a Linux environment for installation using a Windows tool called `winscp`.
3. Even when the binary option is enabled in the `winscp` tool, the installer ZIP used to run and install the agent fails with the above error.

As a workaround, perform the following steps:

1. Download the agent installer to a Linux environment directly and do not transfer it from a Windows environment.
2. If the Oracle Integration user interface is accessible from a Linux environment, use the download install option provided on the Agents page.
3. If the Oracle Integration user interface instance is not accessible, use the following REST command to download the installer to a Linux environment:

```
curl -k -v -X GET -u OIC_user:OIC_password
-H 'Content-Type:application/json'
'https://OIC_URL:443/icsapis/v1/agent/binaries/connectivity'
-o download_location/oic_connectivity_agent.zip
```

Integration Activation Error Due to Change in JDK Location

If integration activation fails with the following error, this is likely the result of the agent installation using a JDK whose location has been changed (for example, removed). This can occur if the agent was installed and running with a version of the JDK whose location was removed and a newer version was installed in a different location. If the JDK installed with the agent is removed, ensure that you restart the agent with the newer version (and location) of the JDK.

```
Caused by: java.lang.Error: Circular loading of installed providers
detected at
java.nio.file.spi.FileSystemProvider.installedProviders(FileSystemProvider.jav
a:161)
at java.nio.file.FileSystems.newFileSystem(FileSystems.java:324)
at java.nio.file.FileSystems.newFileSystem(FileSystems.java:276)
at
oracle.cloud.cpi.agent.store.StoreUtils.extractZipBundle(StoreUtils.java:49)
at
oracle.cloud.cpi.agent.store.FileAgentStore.addActivation(FileAgentStore.java:
40)
at
oracle.cloud.cpi.agent.ActivationCpiCommand.doRequest(ActivationCpiCommand.jav
a:80)
```

Troubleshoot Network Connectivity Issues

Ensure that network connectivity is working correctly when persistent connectivity failures are encountered with an Oracle Integration instance.

 **Note:**

This does not apply to any intermittent failures because the connectivity agent is resilient to temporary conditions and recovers when the situation is resolved.

- Run the following command:

```
nslookup hostname
```

- Run the following command for a period of five minutes to also capture any transient failures:

```
while true;
do
curl https://Oracle_Integration_hostname/ic/home >> file.txt 2>&1;
sleep 1;
echo "trying again";
done;
```

HTTP 401 Unauthorized Error Occurs During Connectivity Agent Installation

If connectivity agent installation or restart fails with an HTTP 401 Unauthorized error, download the agent configuration file from Oracle Integration 3 and restart the connectivity agent.

Troubleshoot Private Endpoints

Get help troubleshooting issues with private endpoints.

Error When Creating a Private Endpoint

The error message that appears on the Log messages page helps you troubleshoot. Which error message did you see?

 **Tip:**

You can open the Log messages page from the Private endpoints page. In the left menu, below Resources, click **Work requests**. Then, in the table of work requests, find the work request for the private endpoint, and click the value in the **Operation** column.

Error	Reason for the error and next steps
Unable to add or enable Private Endpoint Outbound Connection for the integration instance. This error can occur, if the customer tenancy is not configured with the required policy to enable the Private Endpoint. Update or add the policy, and retry.	The error occurs when the policy is missing or incorrect, or you're not part of the group that is assigned to the policy. Check your policy, and make sure you're part of the group that is allowed to create private endpoints. See Prerequisites for Configuring a Private Endpoint.

Error	Reason for the error and next steps
Unable to add Private Endpoint. The option to add a Private Endpoint for the integration instance is not supported.	The error occurs when your region doesn't support private endpoints. If this message appears, you can't create or use private endpoints.
Unable to add Private Endpoint outbound Connection for the Integration instance. This error can occur when all available IP addresses of the Subnet has already been allocated; ensure there are sufficient unassigned IP addresses available in the SubnetId <i>subnet_id</i> .	The error occurs when all the subnet's available IP addresses have already been allocated. Increase the CIDR limit for the subnet.

Unable to Edit a Private Endpoint

You cannot edit an existing private endpoint. However, you can delete the endpoint and create another one.

See [Delete a Private Endpoint](#).

Error When Testing a Connection

The error message that pops up after you test the connection helps you troubleshoot. Which error message did you see?

Error	Reason for the error and next steps
Unable to fetch the value of dnsProxyIp , make sure the endpoint is connecting to a valid private endpoint	The error occurs when the private endpoint hasn't been created yet, or when the private endpoint has been deleted. See Configure a Private Endpoint for an Instance .
CASDK-003: Unable to parse the resource, <i>connection_url</i> . Verify that URL is reachable, can be parsed and credentials if required are accurate.	The error occurs when you select Public gateway as the Access type when creating the connection. Select Private endpoint instead.

Error	Reason for the error and next steps
<pre>{"detail":""."status":"HTTP 500 Internal Server Error","Operation (testConnection) failed: Error while performing AddressTranslation for private endpoint. Please check if the connection url is valid","type":"https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.5.1") or {"detail":""."status":"HTTP 500 Internal Server Error","Operation (testConnection) failed: This connection is configured with an address which is not in an allowable range","type" "https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.5.1"}</pre>	<p>This error can occur for the following reasons:</p> <ul style="list-style-type: none"> When you try to connect to a resource that's in a different VCN than the private endpoint connects to. A private endpoint can connect to a single subnet in a single VCN. Ensure that all private resources are in the subnet that the private endpoint connects to. When you provide a host IP address on the Connections page. To connect through a private endpoint, you should always provide the host FQDN and not the IP address. When the private endpoint is configured in a subnet of a VCN and the endpoint (database/REST service) exists in another subnet whose security list is not open to the private endpoint subnet for access, even though both are on the same VCN. <p>See Security Lists.</p> <p>Open the ingress of the endpoint subnet CIDR to let the private endpoint subnet access the endpoint. Also, the egress of the private endpoint subnet should allow access to the endpoint CIDR. Ensure the network is open between the two subnets.</p> <ul style="list-style-type: none"> Your DNS zone is unable to resolve the Autonomous Transaction Processing custom host name. To resolve this issue, update your DNS zone and add an entry that maps the Autonomous Transaction Processing FQDN to the Autonomous Transaction Processing private IP.

Runtime Errors

Error	Reason for the error and next steps
Error while performing AddressTranslation for private endpoint, Please check if the connection url is valid;	<p>The error occurs when the private endpoint hasn't been created yet, or when the private endpoint has been deleted.</p> <p>See Configure a Private Endpoint for an Instance.</p>

Troubleshoot Integration Runtime

Learn about troubleshooting integration runtime in Oracle Integration.

Topics:

- [An Integration Runs Slowly](#)
- [Fewer Integration Instances Ran Than Usual](#)
- [Using Basic Authentication to Call Services When Oracle Identity Cloud Service is Configured With Multi-Factor Authentication Causes a 401-Unauthorized Error](#)
- [Perform Remedial Actions When the Payload Exceeds the Size Limit](#)

- Remove Unprintable Control Characters from the Incoming Payload
- Notification Action Returns a <from> value is invalid Error
- Synchronous Integration Timeout Failure
- Activity Stream Details
- Message Dequeuing Takes Three to Four Seconds in Asynchronous Integrations
- 403 Error When Accessing an API Protected Using OAuth
- Troubleshoot Certificate Import Failures

An Integration Runs Slowly

Several common issues can result in performance issues for an integration. For example, when the target application takes longer than expected to respond, the integration also runs slower than expected.

How to Determine If Your Integration Is Running Slowly

See [Determine an Integration's Typical Processing Times](#), and then take the appropriate action:

- If the integration has always run slower than expected, determine whether it contains a common pitfall.
See [Design Best Practices](#).
- If the integration has detectable performance outliers, follow the troubleshooting steps on this page.

Step 1. Investigate the Target Application

The target application for an invoke connection might be taking longer than usual to respond. **A slow target application is the most common cause of slow-running integrations.**

To determine whether the target application caused the slowness for your integration, check the logs of the application. The logs help you determine whether the application was experiencing an issue when Oracle Integration connected to it.

Step 2. Check the Number of Incoming Messages

When an integration receives a large number of messages to process, Oracle Integration adds the messages to a queue and processes them until it clears the queue. A long queue can impact the overall performance of your system. For example, a retailer might experience higher message volume during a promotion.

To see how many messages Oracle Integration processed in the last hour:

1. In the navigation pane, click **Observability**, then **Integrations**.
2. Click **Filter** , and update the following filtering requirements:
 - **Time window:** Last 1 hour.
 - **Status:** All.
3. Click **Apply**.
4. Review the numbers in the **Processed** column.

Keep in mind that the number of messages is just one factor in performance. Smaller messages that contain very large incoming payloads can also impact performance.

Step 3. Determine Whether the Message Counts Are Too High

1. Work with an administrator to check whether your organization purchased enough message packs for the messages that you're using. See *Edit the Edition, License Type, and Message Packs of an Instance* in *Provisioning and Administering Oracle Integration 3*.
2. If the integration is synchronous, determine whether the integration is trying to process too many requests at a time. See *Calculate Requests Per Second* in *Provisioning and Administering Oracle Integration 3*.

Step 4. Check for a Clustered Schedule (Schedule Integrations Only)

Schedule integrations can run slower than expected when too many integrations start at the same time.

To determine whether your schedule integrations are clustered together, see [View the Calendar of Schedule Integration Runs](#).

Step 5. Look For Inefficiencies

If the integration processes records, you might be able to fine-tune the efficiency of the processing.

Action for processing records	Recommendations
For-each action	<p>You typically can process 10,000 or fewer records using the for-each action without experiencing performance issues.</p> <p>To process 10,000 or more records, use a for-each action within a parallel action. The parallel action allows you to process the records in parallel threads, thereby improving the performance of the integration.</p>
Stage file action	<p>You typically can read and write 10,000 or fewer records within a file using the stage file action without experiencing timeouts.</p> <p>To process a file with 10,000 or more records, use a stage file action within a parallel action. The parallel action allows you to read and write the records in chunks, thereby improving the performance of the integration.</p>

Haven't Solved the Problem?

Get help from Oracle by entering a service request.

Fewer Integration Instances Ran Than Usual

Consider a scenario in which 100 integration instances typically run by a specific time of day, but only 80 integration instances have run by that time today. You need to quickly determine what happened with your missing integration instances.

Note:

The Observability pages show information about the integrations that are running and have already run, not any missing instances. To identify whether any integration instances are missing, you must know the integrations that usually run.

Why Integrations Don't Run

An integration might not run for a number of reasons. For example:

- Someone deactivated an integration.
- The external application that triggers an integration is down.
- Someone changed the schedule for a schedule integration, such as changing it from running hourly to running daily.

To Troubleshoot Missing Integration Instances:

1. Determine how many integration instances have run so far.
 - a. In the navigation pane, click **Observability**, then **Instances**.
 - b. Click **Filter** , and update the filtering requirements for the page.
For instance, update the time window to the last 1 hour, 6 hours, or 1 day.
 - c. At the top of the page, review the count of the number of integration instances that ran.
2. If fewer integration instances have run than expected, troubleshoot the missing instances.
For instance:
 - Check whether the integrations are active.
 - Check whether an external application that triggers an integration is down.
 - Check whether someone changed the schedule for a schedule integration.
 - If needed, consult the audit trail to determine who last updated an integration.See [Check the Audit History for an Integration or Other Component](#).

Using Basic Authentication to Call Services When Oracle Identity Cloud Service is Configured With Multi-Factor Authentication Causes a 401-Unauthorized Error

If Oracle Identity Cloud Service is configured with multi-factor authentication (for example, user name / password + multi-factor authentication) and you use basic authentication to invoke Oracle Integration services, you receive a 401-Unauthorized error. None of the adapters or the connectivity agent in Oracle Integration support multi-factor authentication.

As an alternative:

- Disable multi-factor authentication from the account with which you are using basic authentication.
- Use OAuth instead.

See [Managing Multi-Factor Authentication](#).

Perform Remedial Actions When the Payload Exceeds the Size Limit

When the message payload in an integration exceeds the size limit, the message is rejected and an error message is displayed. The error message describes remedial actions to perform.

Issue	Error Message	Remedial Action to Perform
Payload size limit exceeded	<ul style="list-style-type: none"> When REST, File, SOAP, FTP, and SAP Adapter connections that support payload sizes of up to 50 MB exceed the limit: <p>CASDK-0059: Unable to process the response received from the client application. The response received by the OIC integration flow is rejected. The response received has been rejected since the content received of length 5723727000 bytes exceeds the maximum allowed threshold of 52428800 bytes (50MB).</p> When all other adapter connections that support payload sizes of up to 10 MB exceed the limit: <p>CASDK-0059: Unable to process the response received from the client application. The response received by the OIC integration flow is rejected. The response received has been rejected since the content received of length 1723727000 bytes exceeds the maximum allowed threshold of 10485760 bytes (10MB).</p> 	The preferred way to send large data sets is by using attachments. Check whether the target application exposes a SOAP API to return data sent as an attachment. If yes, the integration can be designed to configure an invoke connection using the SOAP Adapter to accept an MTOM attachment. The attachment can be an archive containing an XML/comma-separated value (CSV) document.
Attachment size limit exceeded	CASDK-0059: Unable to process the response received from the client application. The response received by the OIC integration flow is rejected. The response received has been rejected since the content received of length 1073741824 bytes (1GB).	Check whether the attachment can be broken and sent through multiple APIs. For example, if the attachment is a ZIP archive containing a large CSV file and unstructured documents, have one API that returns the CSV file and another API that retrieves the unstructured documents.

For additional details about 50 MB payload support, limits, and best practices, see Service Limits in *Provisioning and Administering Oracle Integration 3*.

Remove Unprintable Control Characters from the Incoming Payload

Oracle Integration does not support certain non-printable control characters in the following range.

```
['\u0000', '\u0001', '\u0002', '\u0003', '\u0004', '\u0005', '\u0006',
'\u0007', '\u0008', '\u000b', '\u000c', '\u000e', '\u000f', '\u0010', '
```

```
\u0011', '\u0012', '\u0013', '\u0014', '\u0015', '\u0016', '\u0017', '\u0018', '\u0019']
```

The presence of these control characters in the incoming payload results in the following error during processing:

```
{
  "type" : "10.4.1",
  "title" : "exception occurred while translating the request from native to
xml.",
  "detail" : "oracle.cloud.connector.api.CloudInvocationException:
ORABPEL-15235\n\nTranslation Failure.\nFailed to translate JSON to XML.
org.codehaus.jackson.JsonParseException: Illegal unquoted character
((CTRL-CHAR, code 1)): has to be escaped using backslash to be included in
string value\n at [Source: java.io.BufferedReader@.#; line: 2, column:
22]\nThe incoming data does not conform to the NXSD schema. Please correct
the problem.\n",
  "o:errorCode" : "TRANSLATION-ERROR-00",
  "o:errorDetails" : [ {
    "type" : "NA",
    "instance" : "NA",
    "title" : "NA",
    "o:errorPath" : "NA",
    "o:errorCode" : "NA"
  } ]
}
```

If you receive this error, contact Oracle Support for information on how to preprocess incoming JSON payloads and filter unprintable control characters.

Notification Action Returns a <from> value is invalid Error

If the notification action in an integration validates successfully, but the following error occurs during integration runtime, the value specified in the **Parameter Value** field of the notification action may be incorrect.

```
:
summary=<summary>The <from> value is invalid.
Xpath expression associated with <from> in copy assign activity is invalid.
There is an user error that results in missing element value(s) in the xpath
query.
Please review the payload and modeling to ensure that all elements defined in
the <from> xpath query have valid non-null values .
</summary>
:
,query=<query>fn:current-dateTime()</query>
```

The error can be confirmed by viewing errors for the integration instance on the Instances page or the response of a REST testing tool such as Postman.

Verify that the value specified in the **Parameter** section of the notification action is correct. For example, this value may require conversion to a string:

```
string(fn:current-dateTime())
```

Synchronous Integration Timeout Failure

If a synchronous integration fails to complete within 300 seconds, the following error appears in the activity stream that is viewable from the Instances page or Errors page.

```
CompositeException in:  
io.reactivex.internal.operators.single.SingleOnErrorReturn$OnErrorReturn.onError  
or2 exceptions occurred
```

No details are returned when the call is processed (for example, no 502 status error is returned).

Optimize your integration and/or the endpoint you invoke to complete instance processing in under 300 seconds.

See Service Limits in *Provisioning and Administering Oracle Integration 3*.

Activity Stream Details

You can enable tracing with the payload, view the activity stream logs on the Instances page, and download the activity stream logs with the REST APIs.

- [Enable Tracing with the Payload and View the Activity Stream on the Instances Page](#)
- [Download the Activity Stream Logs with the REST APIs](#)

Enable Tracing with the Payload and View the Activity Stream on the Instances Page

Documentation is available that describes how to enable tracing with the payload and view the activity stream for integrations on the Instances page:

- See [Manage Tracing Levels on Integrations](#)
- See [Activate an Integration](#)

Download the Activity Stream Logs with the REST APIs

Documentation is available that describes how to download activity stream logs in Oracle Integration or with the REST APIs:

- See [Retrieve Activity Stream Messages](#)
- See [Download a Log File](#) (by specifying the `icsflowlog` parameter)

Note:

If you decide to use absolute values (10 MB, instead of a certain size), downloading the file downloads the latest 10 MB file only.

Message Dequeuing Takes Three to Four Seconds in Asynchronous Integrations

You may observe that Oracle Integration sometimes takes three to four seconds to go from Message Received:Instance created and enqueued for processing to message dequeued and processed by Trigger *trigger_name*. This increases the end-to-end processing time of a particular instance.

This is the expected behavior with asynchronous requests. Asynchronous requests are processed using a queue. The first two entries in the flow logs show the enqueue and dequeue times. Therefore, a few seconds lag is expected.

403 Error When Accessing an API Protected Using OAuth

OAuth enables Oracle Integration to access a user's resources on their behalf. REST and other REST-based adapters (such as Google Calendar Adapter, Microsoft Office 365 Outlook Adapter, and others) often access APIs that are protected using OAuth. If these calls result in a 403 error (forbidden), it usually indicates a lack of permissions to access the API.

You need to carefully examine your OAuth configuration and work with the target API owner/administrator to resolve this issue. There may be several conditions leading to this, but the exact cause is best described by the authorization server. Some probable causes of 403 errors are as follows:

- The access token was procured for a scope that does not cover the API being accessed.
- During the phase to provide consent, the user that provided the consent is not the resource owner or owner of the API being accessed.
- The access token expired or is not valid to access the resource any longer.
- The authorization server revoked access privileges to a particular resource.

Troubleshoot Certificate Import Failures

The certificate that you receive from a Certificate Authority (CA) such as Verisign, Entrust, or others can fail during import into Oracle Integration with the following error in the log file:

```
java.security.cert.CertificateException: Unable to initialize,  
java.io.IOException: extra data given to DerValue constructor
```

To resolve this issue, either:

- Obtain a fixed certificate from the CA.
- Convert the certificate to a Distinguished Encoding Rules (DER) binary certificate using tools such as openssl. For example:

```
openssl x509 -outform der -in sfdc-client.crt -out sfdc-client.der
```

Troubleshoot Integration Design Time

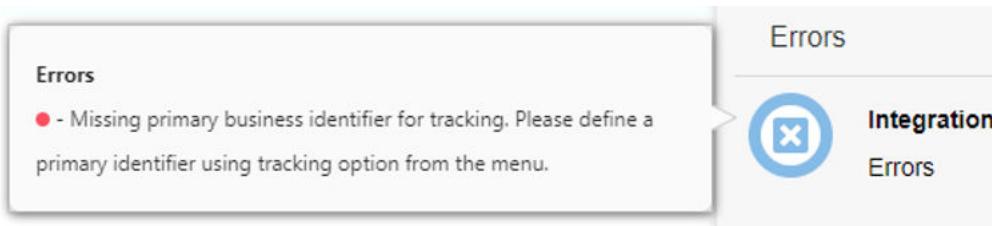
Learn about troubleshooting integration design time in Oracle Integration.

Topics:

- Missing a Primary Business Identifier in an Integration
- Unable to Call a replaceAll JavaScript Function
- Conflict with Using Variables of String Data Type and Applying Mathematical Comparator Expressions
- Reduce Network Calls that Slow Down Processes Caused By Writing Record By Record
- Invoke Subprocess Integrations Using a Single REST API Connection
- Trigger Connection Cannot Receive an Empty Array in JSON if the Corresponding JSON Received Has an Empty JSON

Missing a Primary Business Identifier in an Integration

You must configure at least one business identifier tracking field to make your integration ready for activation. This is known as the primary business identifier. Business identifiers enable you to track payload fields in messages during runtime. If you have not yet configured at least one business identifier, the following error message is displayed when you hover your cursor over the **Integration Errors** icon in the design canvas of your integration.



Configure at least one business identifier to remove this error message. See [Assign Business Identifiers](#).

Unable to Call a replaceAll JavaScript Function

If you attempt to call a `replaceAll` JavaScript function in Oracle Integration, you receive the following error:

```
ode=<code>JavaScript execution error</code>\n,detail=<detail>Review  
JavaScript error(s) and fix them:\nerror: null@3 -> TypeError: Cannot find  
function replaceAll in object
```

The `replaceAll` function is not supported by the JavaScript engine in Oracle Integration.

Use one of the following alternatives to achieve `replaceAll` functionality:

- Use the `replace` function with the global option, as follows:

```
function myFunction(Input) {  
    var res = Input.replace(/<\/>/g, '');
```

```
    return res;
}
```

- Use the `split` and `join` functions instead of `replaceAll`.

Conflict with Using Variables of String Data Type and Applying Mathematical Comparator Expressions

All variables created in Oracle Integration are currently only string data types. Using these variables directly and applying mathematical comparator expressions results in unusual behavior.

For example, in an assign action, two variables, `var1` and `var2`, are created with values of 1 and 10, respectively. In a while action, if you then set the condition as `$var1 < $var2`, the loop terminates after two iterations. The correct usage is to set the condition by wrapping the variables with a `number` XPath function: `number($var1) < number($var2)`. This loop iterates 10 times before terminating.

Reduce Network Calls that Slow Down Processes Caused By Writing Record By Record

Writing record by record using the **Append to Existing File** option on the Operations page of the Adapter Endpoint Configuration Wizard of the FTP Adapter creates too many network calls and eventually slows down the process. Instead, use a stage file action and write the records to a stage directory. Once all records are written, use the **List File** option in the stage file action and the FTP Adapter **Write File** option to transfer the file to an FTP location. This approach reduces processing time and prevents too many calls to the FTP server.

Invoke Subprocess Integrations Using a Single REST API Connection

Assume you follow a modular design approach in which you create separate integrations (think of them as subprocesses) to perform specific tasks. For example:

- File - Staging
- Translation
- Stage - Core
- Trigger Job Processes

Each of these subprocesses is created as a WSDL connector. However, having to maintain multiple connectors complicates the situation. Creating a single REST API and triggering subprocesses using API calls is ideally easier to maintain. Note the following details:

- You can invoke subprocesses using a single REST API connection.
- You can use a schedule now API to trigger the required integration. However, this is only possible if this is a schedule integration. In addition, the schedule now API runs the specified integration immediately and this type of integration cannot be configured to take input parameters.

As an alternative, you can create an application integration with a REST connection used as the trigger that can be configured to take input parameters. That integration can be invoked as

a REST API itself, without the need for the schedule now API to invoke it. The URL for the schedule now API in the v2 version of the REST API is:

```
http://host:port/icsapis/v2/integrations/SAYHELLO%7c01.00.0000/schedule/jobs
```

That is, the URL contains CODE | VERSION to specify which integration to run.

Note that a SOAP connector is the only option available that can call subprocesses.

Trigger Connection Cannot Receive an Empty Array in JSON if the Corresponding JSON Received Has an Empty JSON

An adapter does not receive any element from the mapper if the element is an empty array. To get the element from the mapper in case of an empty array, change the XSL file manually.

1. Export the integration.
2. Manually change the mappings in the XSL file to ensure that the default elements are generated in the target even when the source elements do not exist or are empty. For example:

Before editing the XSL file:

```
<xsl:for-each
select="$Cheers_API/nsmpr0:executeResponse/nsmpr1:response-wrapper/nsmpr1:
employeeEarnedDetails/nsmpr1:earnedBadges" xml:id="id_24">
<nstrgdf1:earnedBadges xml:id="id_23">
<xsl:value-of select="." xml:id="id_25" />
</nstrgdf1:earnedBadges>
</xsl:for-each>
```

After editing the XSL file:

```
<nstrgdf1:earnedBadges xml:id="id_23">
<xsl:for-each
select="$Cheers_API/nsmpr0:executeResponse/nsmpr1:response-wrapper/nsmpr1:
employeeEarnedDetails/nsmpr1:earnedBadges" xml:id="id_24">
<xsl:value-of select="." xml:id="id_25" />
</xsl:for-each>
</nstrgdf1:earnedBadges>
```

3. Import the integration.

Troubleshoot Schedule Integrations

Learn about troubleshooting schedule integrations in Oracle Integration.

Topics:

- [Schedule Integrations Are Not Running on Time](#)
- [When a Schedule Integration Instance Gets Terminated](#)

Schedule Integrations Are Not Running on Time

Schedule integrations may be queued and not run on time as scheduled for the following reasons.

- There is a running instance of the same schedule integration. No new instance of the same integration is run until the running instance completes.
- The throttling limit for schedule integrations has been reached. No new schedule integrations are run until the number of schedule integrations being run falls under the throttling limit.

 **Note:**

The throttling limit is automatically adjusted based on available system resources.

When a Schedule Integration Instance Gets Terminated

An instance of a schedule integration instance can be terminated by a user or Oracle Integration for the following reasons.

- A user cancels a schedule integration instance.
- A user call reactivates a schedule integration, which causes the currently-queued scheduled instance to terminate.
- Scheduled flow processing exceeds six hours, which causes Oracle Integration to terminate the schedule integration instance.

Troubleshoot Integration Activations

Learn about troubleshooting integration activations in Oracle Integration.

Topics:

- [Troubleshoot Integration Activations](#)
- [Delay in Message Processing of First Instance after Activation](#)

Troubleshoot Integration Activations

If your integration fails to activate, review the following failure issues and corrective actions.

Activation Failure Causes	Corrective Actions
WSDL issues: <ul style="list-style-type: none"> • Duplicate schema issues in the WSDL • WSDL/schema validation failures 	<ol style="list-style-type: none"> 1. Check the WSDL for duplicate schemas. 2. Check that the WSDL is valid. 3. If necessary, correct the WSDL in the application. 4. Reconfigure the integration by refreshing the endpoints. 5. Retry the activation.
Deployment timeout issues.	Retry the activation because the server may be slow to react.
Database issues: <ul style="list-style-type: none"> • Reached the size limit: java.sql.SQLException: ORA-01654: unable to extend index • Database is down. • Connection pool issues: 	<ol style="list-style-type: none"> 1. Check that the database is running and has enough space. 2. If not, the database administrator must increase the database size limit or connection pool size. 3. Retry the activation.
<pre>Could not create pool connection for datasource 'SOALocalTxDataSource'. The DBMS driver exception was: Listener refused the connection with the following error: ORA-12516, TNS:listener could not find available handler with matching protocol stack</pre>	
A connection is down or was modified.	<ol style="list-style-type: none"> 1. Check if the service/application is accessible. If it is down, retry the activation later. 2. If the connection is invalid, update the connection configuration. 3. Reconfigure the integration by refreshing the endpoints. 4. Retry the activation.
Managed server unavailable.	<ol style="list-style-type: none"> 1. The administrator must restart the server. 2. Retry the activation.

Activation Failure Causes	Corrective Actions
<p>If an integration contains too many actions (for example, it has 25 or more stage write actions), activation failure can occur:</p> <pre data-bbox="360 375 878 720">RegularFileObject[/u01/data/domains/ iscs /servers/integration_server1/dc/ 338c94 f1-d3ee-45b9-bbf8-43c9203df851/SCA- INF/ bpel/ StageCurrentGTHROFileForEachBpel/ src/orabpel/ stagecurrentgthrofileforeach bpel/BIN0.java]:51 code too large</pre>	<p>Optimize the integration. For example, split the integration into multiple integrations.</p>

Delay in Message Processing of First Instance after Activation

After activation of an integration, there is a several second delay between when the instance is created and when the message is processed. This is the expected behavior.

On the first request for all integrations, the artifacts and back-end processes are lazy loaded as a first step. This can make for a longer time period spent between the first two steps shown in the activity stream when compared to subsequent invocations. This is the expected behavior for the initial request, and by design.

For example, you may see the following in the activity stream:

```
Sep 3rd 12:43:16.442 PM UTC 2020 Message received: Instance create
Sep 3rd 12:43:23 Message processed by Trigger
```

Note the seven second delay between when the message was received and processed.

A

Oracle Integration Use Cases

Use Oracle Integration in scenarios such as the following.

Topics:

- Manage Oracle Cloud Infrastructure Buckets and Objects from an Integration
- Read Large XML Files Containing Multiple Namespaces
- Use a Stage File Write Operation with an Opaque Schema
- Test REST Adapter Trigger-Based Integrations with Multipart Attachments on the Configure and run page
- Calculate the Sum of a Column or the Count of Rows While Processing Large Files

Manage Oracle Cloud Infrastructure Buckets and Objects from an Integration

You can manage Oracle Cloud Infrastructure Object Storage buckets and objects directly from the integration canvas through use of the **OCI Object storage** action. This section provides high-level use cases on how to create a bucket and upload objects directly from an integration.

- Create an Object Storage Bucket
- Upload an Object

Understand the concepts and satisfy all prerequisites before using this feature. See [Invoke Oracle Cloud Infrastructure Object Storage from an Integration with an OCI Object Storage Action](#).

Create an Object Storage Bucket

1. Add a REST Adapter trigger connection to the integration.
 - a. On the Basic Info page, enter a name, and click **Next**.
 - b. On the Resource Configuration page, enter the following information.

Field	Description
What is the endpoint's relative resource URI	Enter the following: /trigger
What action do you want to perform on the endpoint	Select POST .

Field	Description
Select any options that you want to configure	Scroll to the bottom of the page and select the following options: <ul style="list-style-type: none"> • Add and review parameters for this endpoint • Configure a request payload for this endpoint

- c. On the Request Parameters page, create the following query parameters.

Name	Data Type
bucketName	string
objectName	string

- d. On the Request page, create the following query parameters.

Field	Description
Select the request payload format	Binary
What is the media-type of Request Body?	application/octet-stream

- e. On the Summary page, click **Finish**.

2. Add an **OCI Object storage** action to the integration.

3. Enter a name.

4. Select **Manage buckets**.

5. From the **Select operation** list, select **Create bucket**.

6. Select the compartment in which to create the bucket.

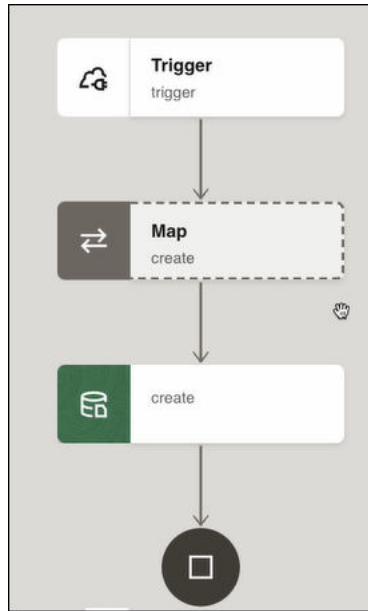
For this example, an object storage bucket is selected for creation in the **oic** compartment in Oracle Cloud Infrastructure.

The screenshot shows a configuration dialog titled "Configure object storage". It contains four stacked dropdown menus. The first menu is labeled "What do you want to call your endpoint?" and has "create" selected. The second menu is labeled "Select the resource type you want to manage" and has "Manage buckets" selected. The third menu is labeled "Select operation" and has "Create bucket" selected. The fourth menu is labeled "Select compartment" and has "oic" selected.

7. Click **Continue**.

8. Review your selections on the Summary page. The provided name, selected compartment, and the OCID for that compartment are displayed.

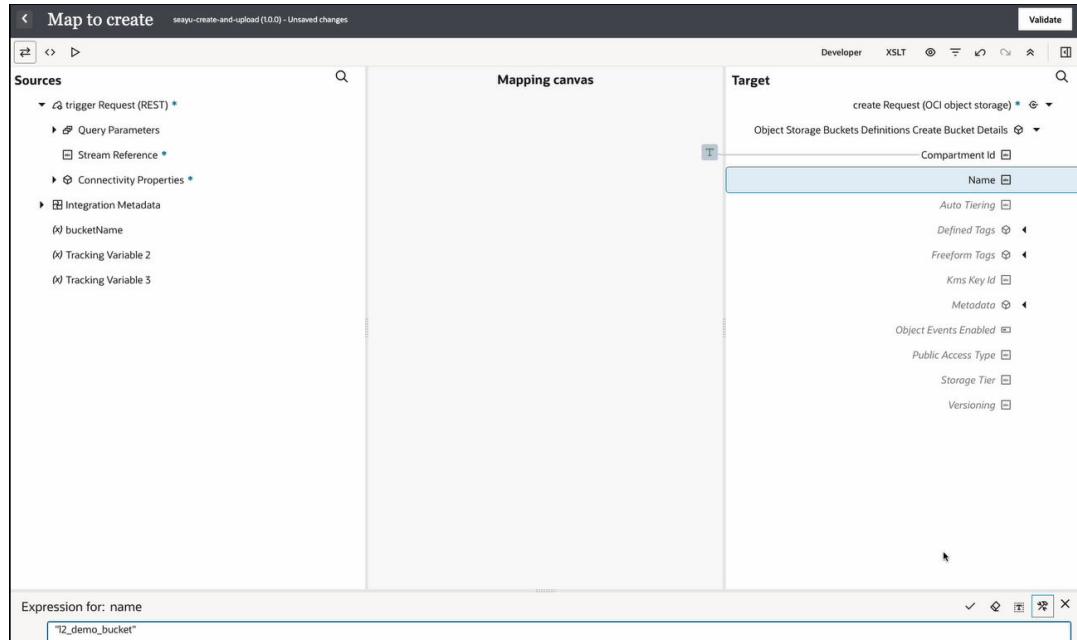
The integration looks as follows.



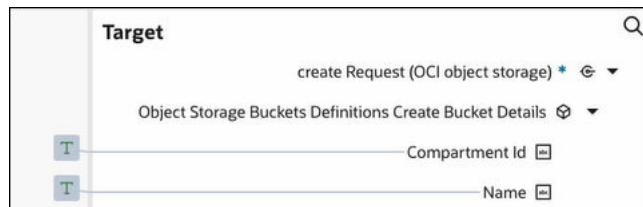
9. Double-click the mapper.
10. In the **Target** section, right-click the **Compartment ID** element and select **Create target node**.
11. In the Expression Builder, specify an OCID value. The value you map to the **Compartment ID** takes precedence over the design-time selection, even if it's a different OCID or a blank value (empty string).

The screenshot shows the 'Map to create' configuration interface. The Sources panel on the left lists various items such as 'trigger Request (REST)', 'Query Parameters', 'Stream Reference', 'Connectivity Properties', 'Integration Metadata', 'bucketName', 'Tracking Variable 2', and 'Tracking Variable 3'. The Mapping canvas is currently empty. The Target panel on the right shows 'create Request (OCI object storage)' with a 'Compartment Id' field selected. Below it, the Expression builder shows the expression 'oci1.compartment.oc1..aaaaaaaaidjsbzxjbeabpfk2bh6k7p5m'. The bottom status bar indicates the expression is for 'nstrgmp:compartmentId'.

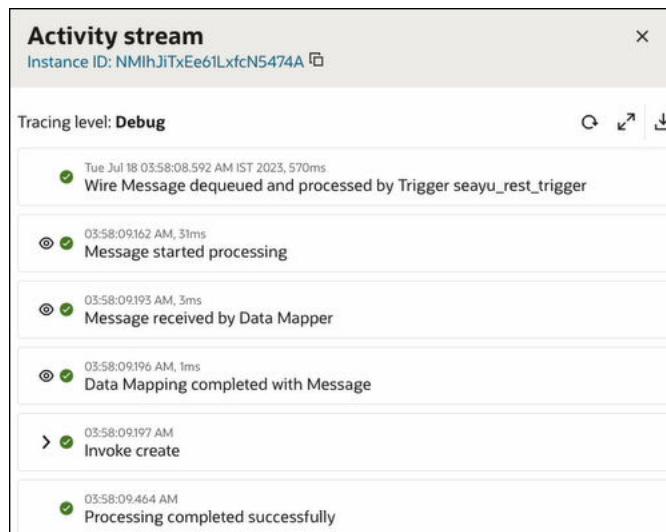
12. In the **Target** section, right-click the **Name** element and select **Create target node**.
13. In the Expression Builder, enter the name to use for the new bucket (for this example, `12_demo_bucket` is entered).



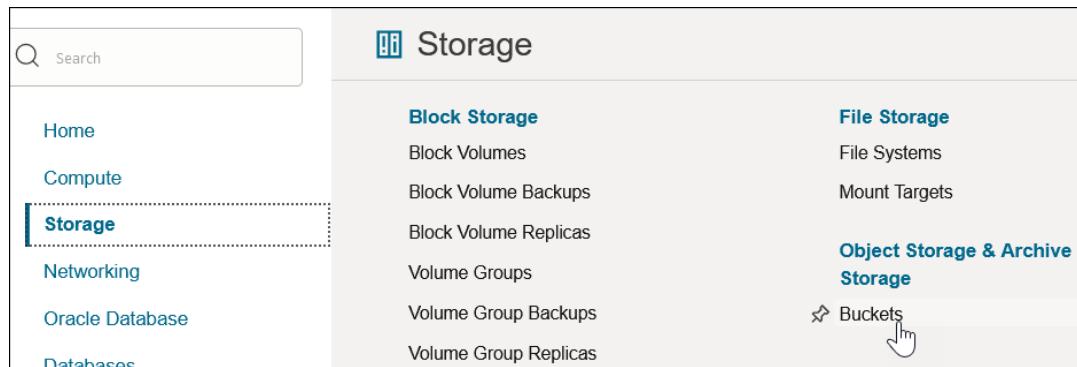
Configuration of the mapper is now complete.



14. Save your changes, and exit the integration.
15. Hover your cursor over the integration, and select **Actions** ..., then **Run**.
16. On the Configure and run page, click **Run**.
17. Review the activity stream.



18. Go to the Oracle Cloud Infrastructure Console.
19. In the navigation pane, select **Storage**, then **Buckets**.



The new bucket (for this example, **I2_demo_bucket**) is visible in the table.

Create Bucket			
Name	Default Storage Tier	Visibility	Created
created-by-rest-adapter	Standard	Private	Thu, Mar 16, 2023, 16:12:41 UTC
created-in-console	Standard	Private	Tue, Mar 14, 2023, 21:06:53 UTC
created-in-I2	Standard	Private	Fri, Jul 14, 2023, 21:57:22 UTC
created-with-postman	Standard	Private	Thu, Mar 16, 2023, 16:12:53 UTC
demo_objects_to_zip	Standard	Private	Mon, Apr 10, 2023, 23:13:40 UTC
demo_zipped_objects	Standard	Private	Mon, Apr 10, 2023, 23:13:47 UTC
I2_demo_bucket	Standard	Private	Mon, Jul 17, 2023, 22:30:18 UTC

Upload an Object

1. Add a REST Adapter trigger connection to an integration.
 - a. On the Basic Info page, enter a name, and click **Next**.
 - b. On the Resource Configuration page, enter the following information.

Field	Description
What is the endpoint's relative resource URI	Enter the following: /trigger
What action do you want to perform on the endpoint	Select POST .
Select any options that you want to configure	Scroll to the bottom of the page and select the following options: <ul style="list-style-type: none"> • Add and review parameters for this endpoint • Configure a request payload for this endpoint

- c. On the Request Parameters page, create the following query parameters.

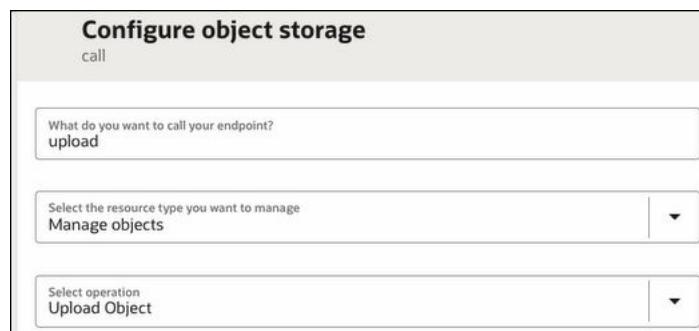
Name	Data Type
bucketName	string
objectName	string

- d. On the Request page, create the following query parameters.

Field	Description
Select the request payload format	Binary
What is the media-type of Request Body?	application/octet-stream

- e. On the Summary page, click **Finish**.

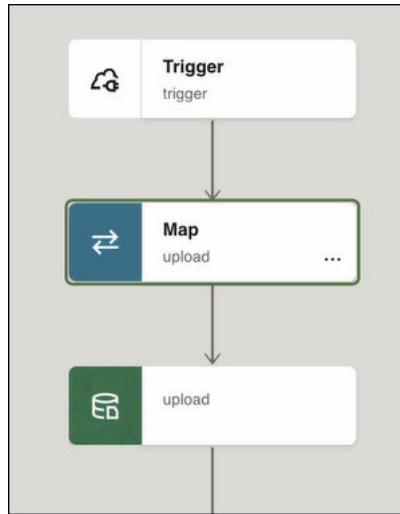
2. Add an **OCI Object storage** action to an integration.
- Enter a name.
 - Select **Manage objects**.
 - From the **Select operation** list, select **Upload Object**, then click **Continue**.



- d. Review your selections on the Summary page, then click **Finish**.
- Click the mapper.
 - Perform the following source-to-target mappings.
 - Map the source **Bucket Name** to the target **Bucket Name**.
 - Map the source **Object Name** to the target **Object Name**.
 - Map the source **Stream Reference** to the target **Stream Reference**.



5. Save the integration.
The integration looks as follows.



6. Exit the integration canvas.
7. Hover your cursor over the integration, and select **Actions** • • •, then **Run**.
8. In the **Request** section, click **File** and upload the object to add to Oracle Cloud Infrastructure (for this example, a jpg file), then click **Run**.

Configure and run
Configure request properties, then run the integration and verify the response. [Click here to read more.](#)

POST
`/ic/api/integration/v1/flows/rest/SEAYU_OBJSTORAGE_TEST/1.0/trigger`

Request

URI parameters	Headers	Body	cURL	Integration properties
<input type="radio"/> Text	<input checked="" type="radio"/> File			
<input type="button" value="Choose File"/> antisocialites.jpg				

9. Go to the Oracle Cloud Infrastructure Console.
10. In the navigation pane, select **Storage**, then **Buckets**.
11. Click the bucket in which you uploaded the object (for this example, **I2_demo_bucket**). The object is displayed in the **Objects** section. Note that the object is renamed from what you uploaded on the Configure and run page.

Bucket Information Tags

General

- Namespace: axyyub
- Compartment: oic
- Created: Mon, Jul 17, 2023, 22:30:18 UTC
- ETag:
- OCID: ...wxhmljfe [Show](#) [Copy](#)

Features

- Default Storage Tier: Standard
- Visibility: Private
- Encryption Key: Oracle managed key [Assign](#)
- Auto-Tiering: Disabled [Edit](#) [?](#)
- Emit Object Events: Disabled [Edit](#) [?](#)
- Object Versioning: Disabled [Edit](#) [?](#)

Usage

- Approximate Object Count: 0 objects [?](#)
- Approximate Size: 0 bytes [?](#)
- Uncommitted Multipart Uploads Approximate Count: 0 uploads [?](#)
- Uncommitted Multipart Uploads Approximate Size: 0 bytes [?](#)

Objects

Upload	More Actions ▾	Search by prefix		
<input type="checkbox"/>	Name	Last Modified	Size	Storage Tier
<input type="checkbox"/>	uploaded_by_oic	Mon, Jul 17, 2023, 22:35:34 UTC	19.18 KB	Standard

Read Large XML Files Containing Multiple Namespaces

You can use a stage file action to read a large XML file that contains multiple namespaces. You upload a ZIP archive file containing the contents. While this use case describes how to use the FTP Adapter with the stage file action, other adapters can receive an XML document as an attachment in the response (for example, the SOAP Adapter, REST Adapter, and Oracle Taleo Enterprise Edition Adapter).

The following XML example shows an export of a candidates list obtained from Oracle Taleo Enterprise Edition. The XML file includes a repeating structure and contains more than one namespace in the document.

```

<Document xmlns="http://www.taleo.com/ws/integration/toolkit/2011/05">
  <Attributes>
    <Attribute name="count">3327</Attribute>
    <Attribute name="duration">0:00:01.443</Attribute>
    <Attribute name="entity">Candidate</Attribute>
    <Attribute name="mode">XML</Attribute>
    <Attribute name="version">http://www.taleo.com/ws/tee800/2009/01</Attribute>
  </Attributes>
  <Content>
    <ExportXML xmlns="http://www.taleo.com/ws/integration/toolkit/2005/07/action/export">
      <record>
        <field name="Number">10160</field>
        <field name="FirstName">Performance</field>
        <field name="LastName">HR Admin</field>
        <field name="EmailAddress">hradmin@invalidmail.com</field>
        <field name="CreationDate">2011-11-16T14:36:14-05:00</field>
      </record>
      <record>
        <field name="Number">10163</field>
        <field name="FirstName">Performance</field>
        <field name="LastName">Manager 1</field>
        <field name="EmailAddress">manager1@invalidmail.com</field>
        <field name="CreationDate">2011-11-16T14:42:10-05:00</field>
      </record>
      <record>
        <field name="Number">10261</field>
        <field name="FirstName">Performance</field>
        <field name="LastName">Matrix Manager</field>
        <field name="EmailAddress">tpmatrix@invalidmail.com</field>
        <field name="CreationDate">2012-03-12T08:28:44-04:00</field>
      </record>
      <record>

```

This use case provides a high-level overview of the steps to read a large XML file with multiple namespaces in segments in an integration. Two FTP Adapters are used to download a large ZIP file from a remote FTP server and write the staging file to a remote FTP server.

1. Add an FTP Adapter to a schedule integration.
2. Configure the FTP Adapter with the **Download File** operation and specify the input directory, file name, and ZIP file download directory.

Operations

Select Operation *

Download File

Select a Transfer Mode *

ASCII Binary

Input Directory

/scratch/stageschemachunk

File Name

Candidate-20221223T145232

Download Directory *

/write/flow

3. Add a stage file action to the integration and configure it as follows:
 - a. On the Configure Operation page, configure the following options:
 - Select **Read File in Segments** from the **Choose Stage File Operation** list.
 - Select **Yes** for **Configure File Reference**.
 - Specify the file reference for the ZIP file downloaded from the FTP server in the **Specify the File Reference** section.
 - b. On the Schema Options page, select **XML schema (XSD) document**.
 - c. On the Format Definition page, configure the following options:
 - Click **Browse** to upload the archive of the schemas (XSDs) that represents the message structure.
 - Select the root element that contains the repeated element.
 - Click the **Edit** icon to select the repeating batch element in the Expression Builder.

The Summary page shows your selections.
4. Add a second stage file action to write the records in comma-separated value (CSV) format.
 - a. On the Configure Operation page, configure the following options:
 - Select **Write File** from the **Choose Stage File Operation** list.
 - Specify the file name of the CSV file in the **Specify the File Name** section.
 - Specify the output directory in the **Specify the Output Directory** section.
 - Select the **Append to Existing File** check box.

- b. On the Schema Options page, select **Sample delimited document (e.g. CSV)**.
- c. On the Format Definition page, configure the following options:
 - Click **Browse** to select the sample CSV file.
 - Specify the record and record set name.
5. Perform the mapping between the two stage file actions.
6. Add a second FTP Adapter to write the staging file to an output directory on a remote FTP site.
7. Design the remaining parts of the integration.

Use a Stage File Write Operation with an Opaque Schema

You can use an opaque schema in a stage file action Read File or Write File operation without concern for a schema for the file. The only condition is that whatever is sent to the opaque element in the opaque schema must be base64-encoded data.

For the following use case:

- The integration downloads a zip file using the FTP Adapter Download File operation and extracts and processes each file.
 - The files are staged temporarily in Oracle Integration using a stage file action List Files operation.
 - The files are iterated over using a for-each action.
 - The files are written to a staged directory using a stage file action Write File operation with an opaque schema.
 - The files are written to an output directory using the FTP Adapter Write File operation.
1. Add a stage file action to an integration. See [Configure a Stage File Action](#).
 2. On the Configure Operation page, select the **Write File** operation and specify the file name and output directory.
 3. On the Schema Options page, select **XML schema (XSD) document**.
 4. On the Format Definition page, make the following selections.
 - a. Select a new file (for this example, **opaque_schema.xsd** is selected).

The contents of the XSD file are as follows for this example:

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<schema targetNamespace="http://xmlns.oracle.com/pcbpel/adapter/opaque/"
       xmlns="http://www.w3.org/2001/XMLSchema" >
  <element name="opaqueElement" type="base64Binary" />
</schema>
```

- b. Select the schema element (for this example, **opaqueElement** is selected).
5. Map the **FileReference** of the for-each repeating element (**ICSFile**) to **opaqueElement** of **StageFileWrite** using an **encodeReferenceToBase64** function.

Test REST Adapter Trigger-Based Integrations with Multipart Attachments on the Configure and run page

You can test REST Adapter trigger-based integrations with multipart attachments on the Configure and run page. This section provides high-level use cases for testing the following multipart attachment types.

- multipart/form-data
- multipart payload with structured data (for this example, JSON)

Test Multipart/Form-Data

1. Configure a REST Adapter as a trigger connection and select the following options on the Request page:
 - **Request is multipart with payload**
 - **Multipart request is of type multipart/form-data with HTML form payload**
2. Design the remaining parts of your integration. For this example, the integration includes a REST Adapter trigger connection, notification action, mapper, and return action.
3. Activate the integration.
4. Hover over the REST Adapter trigger connection-based integration to run.
5. Click **Actions ...**, then select **Run**.
6. In the **URI Parameters** section of the Configure and run page, specify values for the parameters.
7. In the **Headers** section, view the header details.
8. In the **Body** section, select the attachments file to upload and specify a text description for the email.
Click **Run** to invoke this REST Adapter-triggered integration.
The activity stream results are displayed on the right side of the page.
9. Check your response email for the file attachment you selected and the text description you specified.

This document describes oracle rest standards

Test a Multipart Payload with JSON Structured Data

1. Configure a REST Adapter as a trigger connection and select the following options on the Request page:
 - **Request is multipart with payload** as the multipart attachment processing option.
 - **JSON Sample** as the request payload format.
 - **multipart/mixed** or **multipart/form-data** as the media type of the request body.
2. Design the remaining parts of your integration. For this example, the integration includes a REST Adapter trigger connection, notification action, mapper, and return action.
3. Activate the integration.
4. Hover over the REST Adapter trigger connection-based integration to run.

5. Click **Actions . . .**, then select **Run**.
6. In the **URI Parameters** section of the Configure and run page, specify values for the parameters.
7. In the **Body** section, select the attachments file to upload and specify a text description for the email.
8. Click **Run** to invoke this REST Adapter-triggered integration.
The activity stream results are displayed on the right side of the page.
9. Check your response email for the file attachment you selected and the text description you specified.

Oracle rest standards document

The Configure and run page provides a variety of testing features. See [Test Integrations from Outside the Integration Canvas](#).

Calculate the Sum of a Column or the Count of Rows While Processing Large Files

You can process large comma-separated value (CSV) files (up to 1 GB in size) using the Read File In Segments operation of a stage file action. You may also have a requirement to calculate the sum of a column or the count of rows.

Consider the following payload. Assume you want to calculate the sum of all the values in the column **Amount1**. You may typically declare a variable upstream of the stage file action and keep updating this variable with the computation done in each chunk of the stage file action.

Sno	Amount1	Amount2	Amount3	Amount4
Integer	Integer	Integer	Integer	Integer
Man	Man	Man	Man	Man
1	100	100	10.0	600
2	200	100	10.1	600
3	300	100	10.2	100
4	400	100	10.3	200

However, updating the upstream variables inside the stage file action Read File in Segments operation impacts performance and does not allow stage file action processing in parallel. You observe the following warning message in the integration canvas:

Stage File Read File in Segments includes action that will result in segments being processed sequentially

As a solution for this use case, perform the following steps to use the aggregate functions [sum](#) and [count](#) while processing larger files:

1. Add a stage file action in the integration canvas and select the **Read File in Segments** operation to read the large file.
2. Inside the stage file action that uses the **Read File In Segments** operation, create a second stage file action that uses the **Write File** operation and select the **Append to Existing File** check box.
3. Get the count and sum of each segment and write the results into the second stage file action.
 - `count ($StageSegments/.../Record)`
 - `sum ($StageSegments/..Record/ns22:Amount1)`
4. After completion of the stage file action using the **Read File in Segments** operation, create a stage file action that uses the **List File** operation to list the file that was created in Step 2. The response of the stage file action **List File** operation contains the file size. The file size determines the next action to perform.
5. If the file size is less than 10 MB, read the file with a stage file action that uses a **Read Entire File** operation and create an assign action with variables to store the grand totals.
6. Otherwise, repeat Step 1 through Step 4 with the new file.