Oracle® Cloud

Using the OCI Cache Adapter with Oracle Integration 3

G57747-02
September 2026

Oracle Cloud Using the OCI Cache Adapter with Oracle Integration 3,

G57747-02

Copyright © 2026, 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.

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

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Contents

About This Content

1 Understand the OCI Cache Adapter

OCI Cache Adapter Capabilities 1
OCI Cache Adapter Restrictions 2
What Application Version Is Supported? 2
Workflow to Create and Add an OCI Cache Adapter Connection to an Integration 2

2 Create an OCI Cache Adapter Connection

Prerequisites for Creating a Connection 1
Create a Connection 1
- Configure Connection Properties 3
- Configure the Endpoint Access Type 4
- Test the Connection 5

3 Add the OCI Cache Adapter Connection to an Integration

Configure Cache Page 1
Summary Page 3

4 Implement Common Patterns Using the OCI Cache Adapter

Perform Operations in the OCI Cache Cluster 1

Copyright © 2026, Oracle and/or its affiliates.

About This Content

This guide describes how to configure this adapter as a connection in an integration in Oracle Integration.

Audience

This guide is intended for developers who want to use this adapter in integrations in Oracle Integration.

Documentation Accessibility

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

Access to Oracle Support

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

Related Resources

See these Oracle resources:

- Oracle Cloud at `http://cloud.oracle.com`
- *Using Integrations in Oracle Integration 3*
- *Using the Oracle Mapper with Oracle Integration 3*
- Oracle Integration documentation on the Oracle Help Center.

Conventions

The following text conventions are used in this document.

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

Copyright © 2026, Oracle and/or its affiliates.

1

Understand the OCI Cache Adapter

Review the following topics to learn about the OCI Cache Adapter and how to use it as a connection in integrations in Oracle Integration. A typical workflow of adapter and integration tasks is also provided.

Topics:

- OCI Cache Adapter Capabilities
- OCI Cache Adapter Restrictions
- What Application Version Is Supported?
- What Application Version Is Supported?

OCI Cache Adapter Capabilities

The OCI Cache Adapter enables you to invoke an Oracle Cloud Infrastructure (OCI) Cache cluster from an integration in Oracle Integration You can connect to the cache clusters in OCI Cache and use the clusters to manage your data.

OCI Cache is a managed service that enables you to build and manage cache clusters, which are memory-based storage solutions for applications. See About OCI Cache.

The OCI Cache Adapter provides the following capabilities:

- Inserts and reads data from OCI Cache instances. You can cache records as you process data from files/APIs and later merge/summarize the data from the cache.
- Supports operations of the list, set, and string data types. See Configure Cache Page.
- Supports two cluster types:
  - Non-sharded: These clusters can be configured with one to five nodes. One node in the cluster is always configured as the primary node and any other nodes are configured as replicas. All nodes in the cluster are in the same region. However, the service tries to distribute them across fault domains and availability domains (for multiple availability domain regions) as much as possible.
  - Sharded: These clusters don't have primary or replica nodes at the cluster level. These clusters instead have three or more shards, with the data split across the shards in the cluster. Each shard holds part of the data. Each shard has one primary node and up to four replica nodes.
- Supports connecting to private resources that are in your virtual cloud network (VCN) private subnet with a private endpoint. See Configure a Private Endpoint for Your Instance in *Provisioning and Administering Oracle Integration 3* and Configure the Endpoint Access Type. This type of connection does not use the connectivity agent.

You can configure the OCI Cache Adapter as an invoke connection in an integration in Oracle Integration. The OCI Cache Adapter is one of many predefined adapters included with Oracle Integration. See the Adapters page in the Oracle Help Center.

Copyright © 2026, Oracle and/or its affiliates.

OCI Cache Adapter Restrictions

Note the following OCI Cache Adapter restrictions.

- The OCI Cache cluster is accessible only through a private endpoint. The OCI Cache cluster and your Oracle Integration instance must share a private endpoint and the same subnet.
- The **Public gateway** access option is not recommended because it allowlists the subnet and exposes it publicly. Instead, select **Private endpoint** in the **Access type** section of the Connections page. See Configure the Endpoint Access Type.
- You cannot switch between non-sharded and sharded cluster modes in an integration. If you need to change the cluster mode in an integration, create a new OCI Cache Adapter connection. This is a restriction of OCI Cache.

> **Note**
>
> There are overall service limits for Oracle Integration. A service limit is the quota or allowance set on a resource. See Service Limits.

What Application Version Is Supported?

For information about which application version is supported by this adapter, see the Connectivity Certification Matrix.

Workflow to Create and Add an OCI Cache Adapter Connection to an Integration

You follow a very simple workflow to create a connection with an adapter and include the connection in an integration in Oracle Integration.

This table lists the workflow steps for both adapter tasks and overall integration tasks, and provides links to instructions for each step.

Step	Description	More Information
1	Decide where to work	• Work in a project (see why working with projects is preferred in *Using Integrations in Oracle Integration 3*). • Work outside a project.
2	Create the adapter connections for the applications you want to integrate. The connections can be reused in multiple integrations and are typically created by the administrator.	Create an OCI Cache Adapter Connection
3	Create the integration. When you do this, you add trigger (source) and invoke (target) connections to the integration.	Understand Integration Creation and Best Practices in *Using Integrations in Oracle Integration 3* and Add the OCI Cache Adapter Connection to an Integration
4	Map data between the trigger connection data structure and the invoke connection data structure.	Map Data in *Using Integrations in Oracle Integration 3*

Copyright © 2026, Oracle and/or its affiliates.

Step	Description	More Information
5	(Optional) Create lookups that map the different values used by those applications to identify the same type of object (such as gender codes or country codes).	Manage Lookups in *Using Integrations in Oracle Integration 3*
6	Activate the integration.	Activate an Integration in *Using Integrations in Oracle Integration 3*
7	Monitor the integration on the dashboard.	Monitor Integrations During Runtime in *Using Integrations in Oracle Integration 3*
8	Track payload fields in messages during runtime.	Assign Business Identifiers for Tracking Fields in Messages and Track Integration Instances in *Using Integrations in Oracle Integration 3*
9	Manage errors at the integration level, connection level, or specific integration instance level.	Manage Errors in *Using Integrations in Oracle Integration 3*

Copyright © 2026, Oracle and/or its affiliates.

2

Create an OCI Cache Adapter Connection

A connection is based on an adapter. You define connections to the specific cloud applications that you want to integrate.

Topics:

- Prerequisites for Creating a Connection
- Create a Connection

Prerequisites for Creating a Connection

You must satisfy the following prerequisites to create a connection with the OCI Cache Adapter.

Know Your OCI Cache Cluster Connection Properties

You must know the following details to successfully create your OCI Cache Adapter connection. See Configure Connection Properties.

- Know the host of the OCI Cache cluster. Both the cluster mode and host URL are available in the **Connection Details** section of the **Details** tab for the cache cluster in the Oracle Cloud Console.
  1. Log in to the Oracle Cloud Console.
  2. In the left navigation pane, select **Databases**.
  3. Under **OCI Cache**, select **Clusters**.
  4. Click the cluster name to use.
  5. Go to the **Connection Details** section to view the information.
- Know whether you need to use a non-sharded or shared cluster mode.

Create a Connection

Before you can build an integration, you must create the connections to the applications with which you want to share data.

> **Note**
>
> You can also create a connection in the integration canvas. See Define Inbound Triggers, Outbound Invokes, and Actions.

To create a connection in Oracle Integration:

1. Decide where to start:
   - Work in a project (see why working with projects is preferred).

Copyright © 2026, Oracle and/or its affiliates.

a. In the navigation pane, click **Projects**.

b. Select the project name.

c. Click **Integrations**.

d. In the **Connections** section, click **Add** if no connections currently exist or **+** if connections already exist. The Create connection panel opens.

- Work outside a project.

  a. In the navigation pane, click **Design**, then **Connections**.

  b. Click **Create**. The Create connection panel opens.

2. 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.

3. 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. **Note**: *Only* the roles supported by the adapter you selected are displayed for selection. Some adapters support all role combinations (trigger, invoke, or trigger and invoke). Other adapters support fewer role combinations. 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.

Copyright © 2026, Oracle and/or its affiliates.

Element	Description
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.

4. Click **Create**.

   Your connection is created. You're now ready to configure the connection properties, security policies, and (for some connections) access type.

5. Follow the steps to configure 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.

6. Test the connection.

Configure Connection Properties

Enter connection information so your application can process requests.

1. Go to the **Properties** section.
2. In the **Host** field, enter the host of the OCI Cache cluster to invoke. The port number of the cluster defaults to `6379` and cannot be changed.

   When specifying the host, two cluster types are supported.

Type	Purpose	When to Use	Example
`primaryFqdn`	Points to the cache's current DNS primary (writer) node.	Use when your application needs to connect directly to the primary node for write operations or administrative operations.	*host*`-p.redis.`*region*`.oci.oraclecloud.com` The `-p` option represents the primary part for a non-sharded cluster.

Copyright © 2026, Oracle and/or its affiliates.

Type	Purpose	When to Use	Example
`discoveryFqdn`	Provides a stable DNS endpoint that provides cluster topology discovery.	Use for cluster-aware clients that can automatically discover shards, replicas, and topology changes.	*host*-`d.redis.`*region*`.oci.oraclecloud.com` The `-d` represents the discovery part for a sharded cluster.

See Prerequisites for Creating a Connection.

3. In the **Cluster Mode** field, select the mode that corresponds to the host you specified in Step 2:

> **Note**
>
> You cannot mix the cluster mode with the wrong host address. Doing so causes the connection test to fail.

- **Non-Sharded** uses a `primaryFqdn` endpoint because there is only one logical writer node. Therefore, no topology discovery is needed.
- **Sharded** uses a `discoveryFqdn` endpoint because the client must know which shard owns which keys.

For cluster mode descriptions, see OCI Cache Adapter Capabilities.

4. If you want to select the prefix for key isolation, expand **Optional properties**. This capability enables you to isolate access to cache data by selecting how specific a key prefix must be before a user is allowed to access it.
   - **Prefixed with SI**: Enables a user to access cache data in the service instance.
   - **Prefixed with {SI + project_id}**: Enables a user to only access cache data in a specific Oracle Integration project in the service instance.

> **Note**
>
> There are no security policies to explicitly configure with the OCI Cache Adapter.

Configure the Endpoint Access Type

Configure access to your endpoint. Depending on the capabilities of the adapter you are configuring, options may appear to configure access to the public internet, to a private endpoint, or to an on-premises service hosted behind a fire wall.

- Select the Endpoint Access Type
- Ensure Private Endpoint Configuration is Successful

Select the Endpoint Access Type

1. Go to the **Access type** section.
2. Select the option for accessing your endpoint.

Copyright © 2026, Oracle and/or its affiliates.

Option	This Option Appears If Your Adapter Supports ...
Private endpoint	Connections to endpoints using a private virtual cloud network (VCN). **Note**: To connect to private endpoints, you must complete prerequisite tasks in the Oracle Cloud Console. Failure to do so results in errors when testing the connection. See Connect to Private Resources in *Provisioning and Administering Oracle Integration 3* and Troubleshoot Private Endpoints in *Using Integrations in Oracle Integration 3*.

Ensure Private Endpoint Configuration is Successful

- To connect to private endpoints, you must complete prerequisite tasks in the Oracle Cloud 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**.

Test the Connection

Test your connection to ensure that it's configured successfully.

1. 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.
Uses a WSDL	A dialog prompts you to select the type of connection testing to perform: • **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.
   - 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**.

Copyright © 2026, Oracle and/or its affiliates.

3

Add the OCI Cache Adapter Connection to an Integration

When you drag the OCI Cache Adapter into the invoke area of an integration, the Adapter Endpoint Configuration Wizard is invoked. This wizard guides you through configuration of the OCI Cache Adapter endpoint properties.

The following sections describe the wizard pages that guide you through configuration of the OCI Cache Adapter as an invoke in an integration.

Topics:

- Configure Cache Page
- Summary Page

Configure Cache Page

Select the data type and operation to perform.

Element	Description
What do you want to call your endpoint?	Provide a meaningful name so that others can understand the connection. For example, if you are creating a database connection for adding new employee data, you may want to name it `CreateEmployeeInDB`. You can include English alphabetic characters, numbers, underscores, and dashes in the name. You cannot include the following: • Blank spaces (for example, `My DB Connection`) • Special characters (for example, `#;83& or righ(t)now4`) • Multibyte characters
Select data type	• **List**: Stores an ordered sequence of elements. • **Set**: Stores an unordered collection of unique elements. • **String**: Stores a single value (for example, text, number, JSON blob, and serialized object).

Copyright © 2026, Oracle and/or its affiliates.

Element	Description
Select operation	Select the operation to perform. The operations shown are based on the data type you selected. List operations: • **Prepend value(s) to list (LPUSH)**: Adds one or more values to the beginning (left side) of a list. Existing elements are shifted to the right. If the list doesn't exist, it is created automatically. • **Pop first element (LPOP)**: Removes and returns the first element (leftmost element) from a list. If the list is empty or doesn't exist, the operation returns null (or an equivalent `no value` response). • **Append value(s) to list (RPUSH)**: Appends one or more values to the tail end (right side) of a list. If the list doesn't exist, it is created automatically. • **Pop last element (RPOP)**: Removes and returns the last tail element (rightmost element) from a list. If the list is empty or doesn't exist, it returns null (or an equivalent `no value` response). • **Get list length (LLEN)**: Returns the number of elements currently stored in a list. The number of items in a list is counted. The list is not modified. If the list does not exist or is empty, `0` is returned. • **Get list element by index (LINDEX)**: Retrieves a single element from a list stored in a key, without modifying the list. • **Delete list key**: Deletes the specified list and all the elements stored in that list. After the operation completes, the key no longer exists in the cache. Set operations: • **Add member(s) to set (SADD)**: Adds the specified members to the set stored at a given key. If the set does not already exist, it is created automatically. Sets contain unique values only, so duplicate members are ignored. • **Get all set members (SMEMBERS)**: Returns all members contained in the set. Because sets are unordered collections, the returned members may appear in any order. If the key does not exist, an empty set is returned. This operation does not modify the set; it is a read-only operation. • **Remove member(s) from set (SREM)**: Removes the specified members from the set. If a specified member is not present in the set, it is ignored. The number of members that were removed is returned. If the key does not exist, the operation has no impact and returns `0`. • **Delete set key**: Deletes the set key and all members stored in that set. The memory associated with the set is freed. After deletion, the key no longer exists in the cache. The number of keys that were successfully removed is returned. String operations: • **Get data from cache**: Retrieves the value stored for a specified key. The value returned is not modified. If the key does not exist, a `key not found` response is returned. • **Set data into cache**: Stores a value under a specified key in the cache. The key is created if it does not exist. The existing value is overwritten if the key already exists. • **Delete data from cache**: Removes a specified key and its associated value from the cache. After deletion, the key no longer exists. Subsequent attempts to retrieve the key return null (or a `key not found` response). If the key does not exist, the operation usually completes successfully and reports that no key was deleted. • **Check if key exists (EXISTS)**: Determines whether a specified key is present in the cache. Input one or more keys. The number of

Copyright © 2026, Oracle and/or its affiliates.

Element	Description
	those keys that currently exist is output. If a key exists, it is counted. If a key does not exist, it is not counted. • **Set if absent with TTL (SET NX EX)**: Sets a key to a value only if the key does not already exist. A time-to-live (TTL) value is assigned to the key. The expiration time is in seconds.

Summary Page

You can review the specified adapter configuration values on the Summary page.

Element	Description
Summary	Displays a summary of the configuration values you defined on previous pages of the wizard. The information that is displayed can vary by adapter. For some adapters, the selected business objects and operation name are displayed. For adapters for which a generated XSD file is provided, click the XSD link to view a read-only version of the file. To return to a previous page to update any values, click the appropriate tab in the left panel or click **Go back**. To cancel your configuration details, click **Cancel**.

Copyright © 2026, Oracle and/or its affiliates.

4

Implement Common Patterns Using the OCI Cache Adapter

You can use the OCI Cache Adapter to implement the following common pattern.

Topics:

- Perform Operations in the OCI Cache Cluster

Perform Operations in the OCI Cache Cluster

You can perform multiple operations against the OCI Cache cluster with the OCI Cache Adapter.

- Overview
- Design Your Connections
- Design Your Integration
- Activate and Run the Integration

Overview

This use case provides an overview of an integration in which you configure multiple OCI Cache Adapter invoke connections to perform the following operations against the OCI Cache cluster.

Invoke Connection	Data Type	Operation	Description
1	String	Check if key exists (EXISTS)	Checks if a specified key is present.
2	String	Set data into cache	Stores a value under a specified key.
3	String	Get data from cache	Retrieves the value stored for a specified key.
4	String	Delete data from cache	Removes a specified key and its associated value from cache.
5	List	Append value(s) to list (RPUSH)	Appends values to the tail end (right side) of a list.
6	List	Get list length (LLEN)	Returns the number of elements currently stored in a list.
7	List	Delete list key	Deletes a specified list and all the elements stored in that list.
8	List	Get list length (LLEN)	Returns the number of elements currently stored in a list.

Design Your Connections

For this use case, the following connections are created.

Copyright © 2026, Oracle and/or its affiliates.

1. Create and configure a REST Adapter trigger connection to use the OAuth 2.0 security policy.
2. Create and configure an OCI Cache Adapter invoke connection:
   a. Specify the host for the OCI Cache cluster.
   b. Select **Non-Sharded** as the cluster mode.

Design Your Integration

1. Create an application integration.
2. Drag the REST Adapter into the integration canvas and configure it as follows:
   a. Set the endpoint relative response URI. For this example, `/operations` is specified.
   b. Select **GET** as the action to perform on the endpoint.
   c. Select **Configure this endpoint to receive the response**.
   d. Select **JSON Sample** as the response payload format.
   e. Specify sample JSON content:

```
{
  "keyExists" : "false",
  "valueAfterSet" : "test-value",
  "listLenAfterInsert" : 1,
  "listLenAfterPop" : 0
}
```

   f. Select **JSON** as the media-type of the response body.
3. Drag a first OCI Cache Adapter connection into the integration canvas and define the Configure cache page to check if a specified key is present in the OCI Cache cluster.
   a. Enter a name.
   b. From the **Select data type** list, select **String**.
   c. From the **Select operation** list, select **Check if key exists (EXISTS)**.



4. In the map action between the REST Adapter and OCI Cache Adapter, set a value for the target **Key** element. For this example, `"test-key-string"` is the key to check for in the OCI Cache cluster.

Copyright © 2026, Oracle and/or its affiliates.



5. Drag a second OCI Cache Adapter connection into the integration canvas and define the Configure cache page to set a value under a specified key in the OCI Cache cluster.
   a. Enter a name.
   b. From the **Select data type** list, select **String**.
   c. From the **Select operation** list, select **Set data into cache**.



6. In the map action, set values for the target **Value** and **Key** elements. For this example, `"test-key-string"` is the **Key** and `"test-value-string"` is the **Value** to set under the key in the OCI Cache cluster.



Copyright © 2026, Oracle and/or its affiliates.



7. Drag a third OCI Cache Adapter connection into the integration canvas and define the Configure cache page to retrieve the value stored for the specified key in the OCI Cache cluster.
   a. Enter a name.
   b. From the **Select data type** list, select **String**.
   c. From the **Select operation** list, select **Get data from cache**.



8. In the map action, set a value for the target **Key** element. For this example, `"test-key-string"` is the key (and its value) retrieved from the OCI Cache cluster.



9. Drag a fourth OCI Cache Adapter connection into the integration canvas and define the Configure cache page to delete data in the OCI Cache cluster.

Copyright © 2026, Oracle and/or its affiliates.

a. Enter a name.

b. From the **Select data type** list, select **String**.

c. From the **Select operation** list, select **Delete data from cache**.



10. In the map action, set a value for the target **Key** element. For this example, `"test-key-string"` is the key (and its value) deleted from the OCI Cache cluster.



11. Drag a fifth OCI Cache Adapter connection into the integration canvas and define the Configure cache page to append values to the tail end (right side) of a list in the OCI Cache cluster.

a. Enter a name.

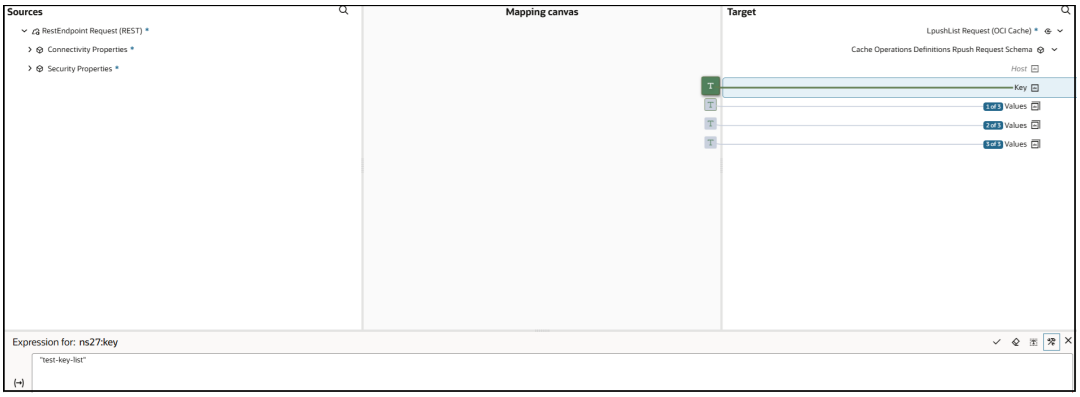
b. From the **Select data type** list, select **List**.

c. From the **Select operation** list, select **Append value(s) to list (RPUSH)**.

Copyright © 2026, Oracle and/or its affiliates.



12. In the map action, set values for the following target elements to append to the end of a list in the OCI Cache cluster.
    - Set **Key** to `"test-key-list"`.
    - Set **(1 of 3) Values** to `"test-value-list-0"`.
    - Set **(2 of 3) Values** to `"test-value-list-1"`.
    - Set **(3 of 3) Values** to `"test-value-list-2"`.



13. Drag a sixth OCI Cache Adapter connection into the integration canvas and define the Configure cache page to return the number of elements currently stored in a list in the OCI Cache cluster.
    a. Enter a name.
    b. From the **Select data type** list, select **List**.
    c. From the **Select operation** list, select **Get list length (LLEN)**.

Copyright © 2026, Oracle and/or its affiliates.



14. In the map action, set a value for the target **Key** element. For this example, `"test-key-list"` is the key of the list for which to check the length in the OCI Cache cluster.



15. Drag a seventh OCI Cache Adapter connection into the integration canvas and define the Configure cache page to delete a specified list and all the elements stored in that list in the OCI Cache cluster.
    a. Enter a name.
    b. From the **Select data type** list, select **List**.
    c. From the **Select operation** list, select **Delete list key**.



16. In the map action, set a value for the target **Key** element. For this example, `"test-key-list"` is the list key to delete in the OCI Cache cluster.

Copyright © 2026, Oracle and/or its affiliates.



17. Drag an eight OCI Cache Adapter connection into the integration canvas and define the Configure cache page to return the number of elements currently stored in a list in the OCI Cache cluster.
    a. Enter a name.
    b. From the **Select data type** list, select **List**.
    c. From the **Select operation** list, select **Get list length (LLEN)**.

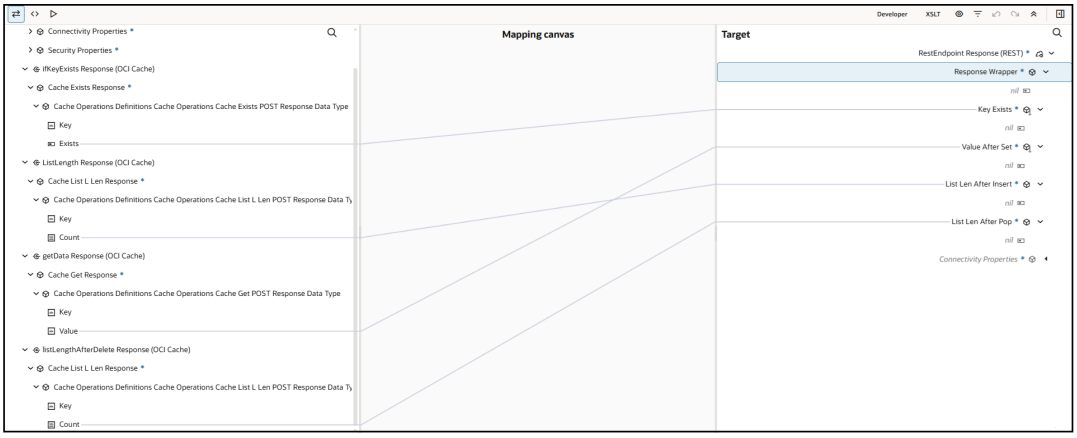


18. In the map action, set a value for the target **Key** element. For this example, `"test-key-list"` is the key of the list for which to check the length in the OCI Cache cluster.



19. In the map action at the end of the integration, map the following source elements and target elements.

Copyright © 2026, Oracle and/or its affiliates.



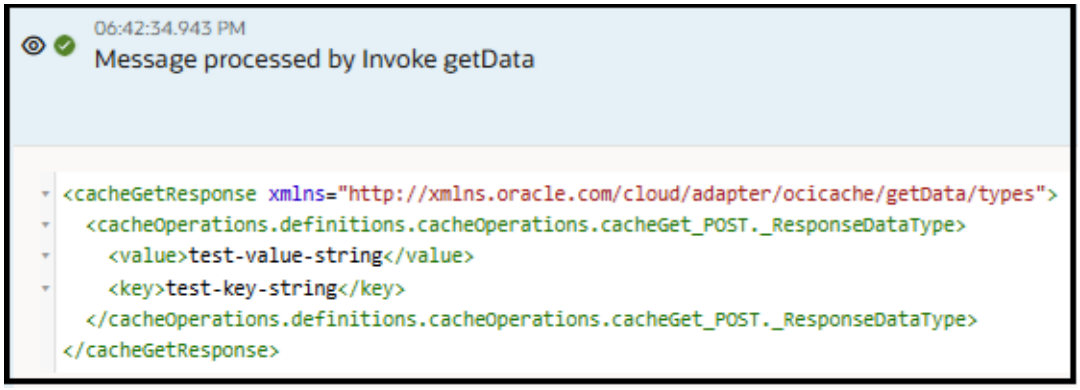
Activate and Run the Integration

1. Assign a business identifier to the integration, then activate it.
2. From the **Actions** ⋮ list for the integration, select **Run**.
3. Click **Run**.
4. Expand the activity stream to view key milestones. Several milestones are described below.
5. View the key being provided to the OCI Cache cluster (`test-key-string`).

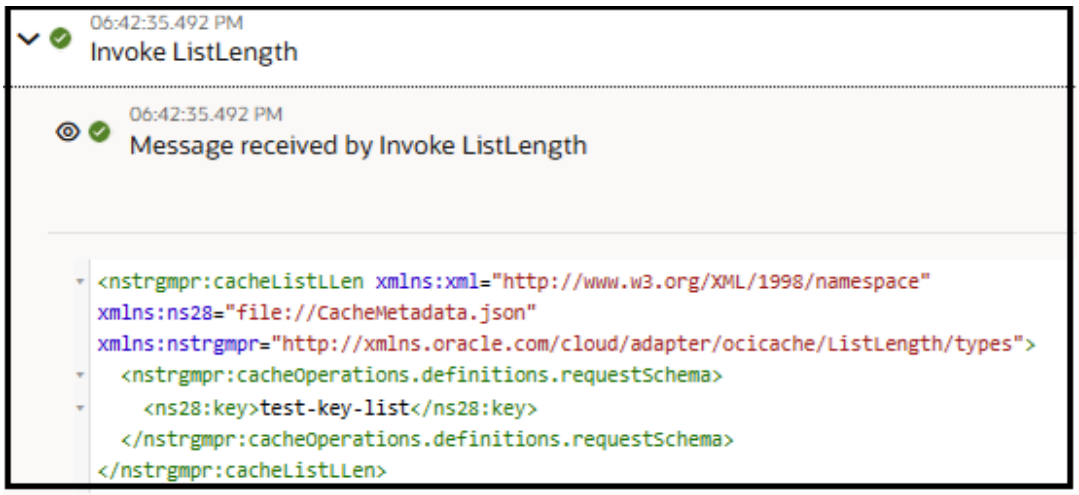


6. View the value and key in the response message (`test-value-string` and `test-key-string`).

Copyright © 2026, Oracle and/or its affiliates.



7. View the list key received (`test-key-list`).

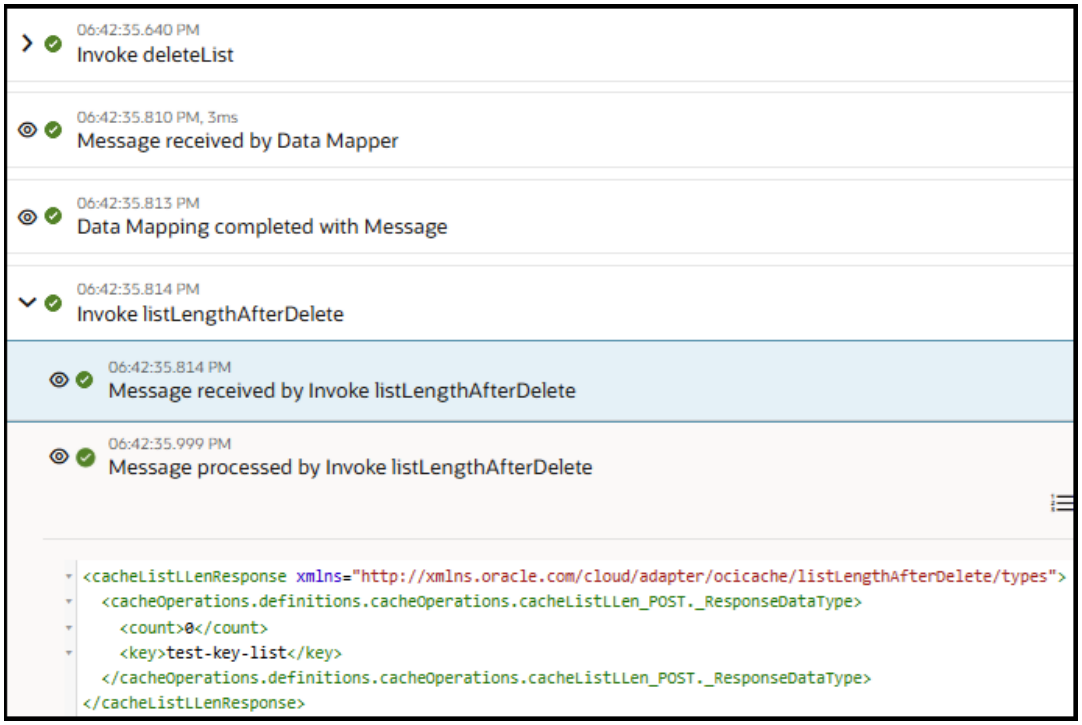


8. View the count of elements in the list that is returned.



9. After the list is deleted, the count returned is zero.

Copyright © 2026, Oracle and/or its affiliates.



Copyright © 2026, Oracle and/or its affiliates.