Oracle® Cloud

Create Activities in Oracle Primavera Cloud for Jira Issues

F70955-02
April 2024

ORACLE®

Contents

Preface

This document describes how to install, configure, and run this recipe in Oracle Integration 3.

Topics:

- Documentation Accessibility
- Diversity and Inclusion
- Related Resources
- Conventions

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.

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

For more information, see these Oracle resources:

- Oracle Integration documentation in the Oracle Cloud Library on the Oracle Help Center.
- Oracle Cloud at `http://cloud.oracle.com`.

Conventions

The following text conventions are used in this document.

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

1
About This Recipe

Use this recipe to create corresponding activities in Oracle Primavera Cloud for issues retrieved from Jira based on a filter ID.

> **✎ Note:**
>
> This recipe is available as **Jira — Oracle Primavera Cloud | Create Activities for Issues** in the Integration Store. Oracle provides this recipe as a sample only. The recipe is meant only for guidance, and is not warranted to be error-free. No support is provided for this recipe.

Overview

This recipe creates activities in Oracle Primavera Cloud for each issue retrieved from Jira based on a filter ID, as per a schedule specified in Oracle Integration.

To use the recipe, you must install the recipe and configure the connections and other resources within it. Subsequently, you can activate and run the integration flow of the recipe manually or specify an execution schedule for it. When triggered, in the first run, the integration flow reads all the issue records in the specified filter, from the Jira instance and, in turn, creates corresponding activity records in your Oracle Primavera Cloud instance. In subsequent runs, it creates activities for new Jira issues (that match the filter) and also propagates updates made in earlier Jira issues to the corresponding Oracle Primavera Cloud activities. Basic data associated with Jira issues, such as issue type, summary, priority, description, status, and comments are synchronized between the two platforms.

System and Access Requirements

- Oracle Integration, Version 21.2.1.0.0 (210129.2200.39462) or higher
- Atlassian
- Oracle Primavera Cloud
- An account on Atlassian with the Administrator role
- An account on Oracle Primavera Cloud with the Administrator role
- At least one Jira project on your Atlassian account. The project(s) can be in any of the Jira products, for example, Jira Work Management or Jira Software
- At least one issue in your Jira project
- At least one project in your Oracle Primavera Cloud account

2

Before You Install the Recipe

You must perform the following configuration tasks on your Atlassian and Oracle Primavera Cloud instances in order to successfully connect to these external systems using Oracle Integration and create activities for issues.

Configure Jira

To access Jira from Oracle Integration, you must perform the following configuration tasks:

1. Create an app, configure the OAuth 2.0 authorization code, and assign the scopes. See Prerequisites to Use the Authorization Code Credentials Security Policy.

2. Create a custom field in Jira to store the Oracle Primavera Cloud activity ID. See Create a custom field. Specify the name of the field as `PrimaveraActivity`.

3. Note the ID of the custom field. See Edit a custom field. When you click **Edit Details**, the Edit Custom Field Details page appears. From the URL displayed in the browser, note the ID available at the end of the URL. For instance, `id=10064`.

4. Create an issue filter in Jira and obtain its ID.

    a. To create a filter, see Save your search.

    b. To view and obtain the filter ID, see Managing your existing filters. Open the filter, and from the URL displayed in the browser, note the ID available at the end of the URL. For instance, `filter=10001`.

Configure Oracle Primavera Cloud

Create a custom field in your Oracle Primavera Cloud project. See Add a Project Field.

3

Install and Configure the Recipe

On your Oracle Integration instance, install the recipe to deploy and configure the integration and associated resources.

1. On the Oracle Integration Home page, in the **Use a recipe/accelerator** section, click **View all**.

2. Find the recipe you want to install, then click **Get**.

   A message confirms that the recipe was successfully installed, and the recipe card shows **In use**.

3. Click **Configure** on the recipe to configure its resources.

   The project workspace opens, displaying all the resources of the recipe. Configure the following resources before you activate and run the recipe.

Configure the Oracle REST Primavera Connection

1. In the Connections section, click the connection name.

2. In the Properties section, enter the following details:

Field	Information to Enter
Connection Type	Leave **REST API Base URL** selected.
Connection URL	Enter your Oracle Primavera Cloud instance URL.

3. In the Security section, enter the following details:

Field	Information to Enter
Security Policy	Select **OAuth Client Credentials**.
Access Token URI	Enter `https://<Primavera_Host_Name>/primediscovery/apitoken/request.`
Client Id	Enter the username of your Oracle Primavera Cloud instance.
Client Secret	Enter the password of your Oracle Primavera Cloud instance.
Scope	Enter `http://{Primavera_Host_Name}/api.`
Client Authentication	Select **Send client credentials as basic auth header**.

4. Click **Save**. If prompted, click **Save** again.

5. Click **Test** to ensure that your connection is successfully configured. In the resulting dialog, click **Test** again.

   A message confirms if your test is successful.

6. To return to the project workspace, click **Go back** .

Configure the Oracle JIRA Connection

1. In the Connections section, click the connection name.

2. In the Properties section, enter the Atlassian host name.

3. In the Security section, enter the following details:

Field	Information to Enter
Security Policy	Select **Authorization Code Credentials**.
Client ID	Enter the integration key of the OAuth application created in Jira. See Configure Jira.
Client Secret	Enter the client secret of the OAuth application created in Jira.
Scope	Enter `read:jira-work, manage:jira-project, write:jira-work, manage:jira-configuration, read:jira-user.`

4. Click **Save**. If prompted, click **Save** again.

5. Click **Test** to ensure that your connection is successfully configured. In the resulting dialog, click **Test** again.

   A message confirms if your test is successful.

6. To return to the project workspace, click **Go back** .

Configure the Lookup Table

Edit the following lookup tables to map the fields between your Jira and Oracle Primavera Cloud instances.

- **ORACLE-BRT-PRIMAVERA_CONSTANTS**: Used to map the fields between Jira and Oracle Primavera Cloud.

- **ORACLE-BRT-JIRA_PRIMAVERA_STATUS**: Used to map the status of the issue in Jira with the status of the corresponding activity in Oracle Primavera Cloud.

1. In the Lookups section, click the lookup name.

2. To configure the **ORACLE-BRT-PRIMAVERA_CONSTANTS** lookup table:

   a. In the **Name** column, enter the name of the Oracle Primavera Cloud field you want to map.

   b. In the **Value** column, enter the name of the field in Jira to which the Oracle Primavera Cloud field has to be mapped.

3. To configure the **ORACLE-BRT-JIRA_PRIMAVERA_STATUS** lookup table:

   a. In the **JIRA_STATUS** column, enter the status of the issue in Jira.

   b. In the **PRIMAVERA_STATUS** column, enter the status of the corresponding activity in Oracle Primavera Cloud.

4. Click **Save**. If prompted, click **Save** again.

5. To return to the project workspace, click **Go back** .

Configure the Integration Flow

Update the integration flow of the recipe (**Oracle Jira Primavera Task Activity Sync**) with the ID of the custom field you created in Jira.

1. In the Integrations section, click the integration flow name.

2. Configure the Switch condition.

    a. On the integration canvas, click the first switch case (**IF customfield_10064**), and then click **Actions** ••• and select **Edit**.

    b. On the resulting page, select **Expression Mode**. In the Source section, search and select the ID of the custom field you created in Jira previously. See Configure Jira.

    c. Click the move button to add the custom field to the expression editor.

    d. Click **Validate**, and then **Close**.

3. Update the data mapping.

    a. On the integration canvas, click the **Map to updateActivity** element, and then click **Actions** ••• and select **Edit** to open the mapper.

    b. In the data mapper, under Source section search for the ID of the custom field you created in Jira previously and map it to **Activity Code** within the **Request Wrapper**.

    c. Click **Validate**, and then **Close**.

4. To return to the project workspace, click **Go back** .

4

Activate and Run the Recipe

After you've configured the connections and other resources, you can activate and run the recipe.

1. In the project workspace, click **Activate**. In the Activate project panel, with the default project deployment selected, choose an appropriate tracing option, then click **Activate**.

   A message confirms that the integration has been activated. Refresh the page to view the updated status of the integration.

2. Run the recipe.

   a. In the Integrations section of the project workspace, click **Actions ...** on the integration flow, then select **Run**.

   b. On the Configure and run page, click **Run**.

   You've now successfully submitted the integration for execution.

   > ✎ **Note:**
   >
   > You can also schedule this integration to run at a date, time, and frequency of your choosing. See Define the Integration Schedule.

3. Monitor the running of the integration flow in Oracle Integration.

   a. In the project workspace, click **Observe**. You'll see the integration flow being triggered and running successfully.

   b. To manage errors in your project, see Manage Errors in a Project.

   In the first run, the recipe creates corresponding activities in Oracle Primavera Cloud for issues retrieved from Jira based on a filter ID. In subsequent runs, it creates activities for new Jira issues (that match the filter) and also propagates updates made in earlier Jira issues to the corresponding Oracle Primavera Cloud activities

4. Check for the new (or updated) activities.

   Log in to your Oracle Primavera Cloud instance. From the menu, select **Schedule** and then select **Activities**.

Related Documentation

- *Using the Jira Adapter with Oracle Integration 3*
- *Using the REST Adapter with Oracle Integration 3*