Oracle Public Sector Permitting and Licensing

Implementing Your Cloud Integrations

24D

Oracle Public Sector Permitting and Licensing
Implementing Your Cloud Integrations

24D

G14839-01

Author: Oracle Permitting and Licensing User Assistance

Contents

Get Help

There are a number of ways to learn more about your product and interact with Oracle and other users.

Get Help in the Applications

Use help icons to access help in the application. If you don't see any help icons on your page, click your user image or name in the global header and select Show Help Icons.

Get Support

You can get support at *My Oracle Support*. For accessible support, visit *Oracle Accessibility Learning and Support*.

Get Training

Increase your knowledge of Oracle Cloud by taking courses at *Oracle University*.

Join Our Community

Use *Cloud Customer Connect* to get information from industry experts at Oracle and in the partner community. You can join forums to connect with other customers, post questions, and watch events.

Learn About Accessibility

For information about Oracle's commitment to accessibility, visit the *Oracle Accessibility Program*. (if videos) Videos included in this guide are provided as a media alternative for text-based topics also available in this guide.

Share Your Feedback

We welcome your feedback about Oracle Applications user assistance. If you need clarification, find an error, or just want to tell us what you found helpful, we'd like to hear from you.

You can email your feedback to *oracle_fusion_applications_help_ww_grp@oracle.com*.

Thanks for helping us improve our user assistance!

2 Working with Oracle Cloud Infrastructure Process Automation

Process Automation Overview

This topic provides an overview of how Oracle Permitting and Licensing uses OCI Process Automation and provides links to the OCI Process Automation documentation that is particularly useful for an Oracle Permitting and Licensing implementation.

Oracle Permitting and Licensing uses OCI Process Automation for designing and processing:

- Workflow
- Fee calculations

You use the Processes module to define:

- Structured processes for Permits, Planing and Zoning, and Business Licenses workflow.
- Dynamic process for Code Enforcement workflow.
- Decision models to calculate fees for your transactions.

OCI Process Automation Documentation and Training

Before you begin implementing workflow and fees using OCI Process Automation, it is *imperative* that you become familiar with the OCI Process Automation features. The documentation in the Oracle Permitting and Licensing guides assume a general knowledge of OCI Process Automation and focus primarily on topics specific to the implementation and doesn't seek to duplicate information covered in OCI Process Automation documentation or training.

For more information on the OCI Process Automation, see *Oracle Cloud Infrastructure Process Automation.*

The link above provides access to valuable information such as:

- Guides
- Recipes (cookbook examples)
- Training

While all of the information available for OCI Process Automation provide by the Oracle Cloud team is useful, in particular these topics are very valuable to help orient you as you being your implementation.

Topic	Description
Overview of Oracle Cloud Infrastructure Process Automation	Provides introductory information about the workspace, runtime, and use cases.
Applications at a Glance	Describes how to use the interface for setting up process applications, decision models, and roles.
Design Structured Processes	Explains how to create and work with structured workflow processes, which are used by Permits, Planning and Zoning, and Business Licenses.

Topic	Description
Design Dynamic Processes	Explains how to create and work with dynamic workflow processes, which are used by Code Enforcement.
Model Decisions	Explains how to create and work with decision models for fee calculations.

OCI Process Automation Transition

Beginning with Release 24B, new customers will begin using OCI Process Automation. Existing customers will continue to use OIC Generation 2 for workflow and fees. At a later date, existing customers will receive notification on when they will switch from OIC Generation 2 to OCI Process Automation. For documentation purposes, existing customers should refer to the Release 24A version of the documentation for any OIC Generation 2 information.

Because the transition from OIC to OCI Process Automation can generate questions, a My Oracle Support page has been created to provide new information as it arises.

You can find updated information regarding this transition on My Oracle Support Document ID: 3005969.1, *Oracle Integration Cloud (OIC) Generation 2 to Generation 3.*

Selecting Your Workflow and Fee Decision Model Source

This topic describes how to select the source component of your workflow definitions and fee model decision definitions (DMN models) during the transition from Oracle Integration Cloud Generation 2 to Oracle Cloud Infrastructure (OCI) Process Automation.

During the transition from Oracle Integration Cloud Generation 2 to OCI Process Automation, existing customers will be configured for Oracle Integration Cloud Generation 2 for workflow and fee processing, while new customers will be on OCI Process Automation for workflow and fee processing.

To set the workflow and DMN model source:

1. Navigate to *Setup and Maintenance*.
2. In Functional Setup Manager, click View Configuration.
3. In the System Administration section, click the Features icon.
4. For the Selecting Workflow and DMN Source feature, click the Features icon.
5. On the Selecting Workflow and DMN Source dialog, select one of these options:
    - *Oracle Integration Cloud:* Indicates OIC Generation 2 will be used for workflow and fee design and processing.
    - *OCI Process Automation:* Indicates OCI Process Automation will be used for workflow and fee design and processing.

Setting Up Workflow and Decision Applications in OCI Process Automation

This topic describes accessing OCI Process Automation from Oracle Permitting and Licensing and provides a brief overview of the user interface.

Accessing OCI Process Automation

You access OCI Process Automation through Functional Setup Manager.

To access Process Automation:

1. Select the *Setup and Maintenance* tile from the agency springboard.
2. Make sure Functional Setup Manager is displaying the appropriate setup template for your offering, such as *Public Sector Permits.*
3. Select the appropriate functional area for your offering.

Offering	Functional Area
Permits	Permit Types
Planning and Zoning	Planning Application Types
Business Licenses	Business License Types
Code Enforcement	Incidents and Cases

4. In the task list, select *Manage Workflow and Decision Applications in OCI Process Automation.*

   This takes you to OCI Process Automation with the Applications node selected, where you can view, create, and modify Process Applications (for workflow) and Decision Applications (for fees).

   > **Note:** The *Applications* node displays all applications, both workflow and fee applications, or you can select *Process* to show workflow or *Decisions* to show fee decision models.

Working with Workflow Applications

When you first access a process application, you'll notice the landing page displays the collection of all the elements that can comprise a process application.

The following table describes the key elements and concepts that make up a workflow application. When you set up a type and you choose the appropriate process definition, you need to specify each of these hierarchical objects.

Object	Description
Application	Applications are functional groups of process definitions containing one or more process definitions. Within an application, you can access a variety of features, including processes (workflow) and decisions (fees). Certain configurations, including integrations and roles, are defined at the application level and shared by all of the application's process definitions. Therefore, you can simplify the setup process by grouping related process definitions into a single application.
Process Definition	A process definition is a specific workflow process. When different transaction types have the same workflow, they can use the same process definition. See *Reviewing a Sample Process Definition* to walk through an example of a process definition for workflow.
UIs	UIs are forms or user interfaces that you can include to display at various points in your workflow if user input is required.
Decisions	Decisions are decision models. Typically in Permitting and Licensing you'll use decisions for calculating fees, but they can be included within workflow processes as well.
Version	When you activate a modified application to make it available for use, you choose a version number to assign. New and modified process definitions can't be associated with a transaction type until you activate a version of the application that includes your changes. For more information on versions, see *Preparing the Process Definition for Use*.
Connectors	There are a collection of integrations called connectors that enable the process definition to share data with the Permitting and Licensing system.
Roles	Roles map to security roles, or groups, in the Permitting and Licensing system, which you assign to swimlanes in your process models. For instance, only certain tasks would be added to the swimlane for the permit technician role, where as others would be added to the permit supervisor role swimlane.

Another concept to keep in mind when working with process definitions is the version. When you activate a modified application to make it available for use, you choose a version number to assign.

New and modified process definitions can't be associated with a transaction type until you activate a version of the application that includes your changes.

When a transaction type is assigned a process application version, new transactions are assigned that version of the application for its life cycle.

For more information on versions, see *Preparing the Process Definition for Use*.

For information on working with workflow for Oracle Permitting and Licensing see *Structured Workflow Overview* and *Dynamic Workflow Overview*.

Working with Decision Applications

When working with fee decision models you:

- Define your input data.
- Configure your decisions.
- Activate the model.

Setting Up OCI Process Automation Proxy User

OCI Process Automation provides the framework for setting up workflow processes. This topic provides information about using the Security Console to set up a proxy user that OCI Process Automation will use for updating workflow information for Oracle Permitting and Licensing.

In this procedure, you create a user and assign the PSCR Proxy User for OIC (CUSTOM_PSCR_OIC_PROXY_USER) role to that user. The user you create is the proxy user the OCI Process Automation system uses to connect to Oracle Permitting and Licensing to exchange data during transaction processing (callback). To complete these steps, you will use Fusion Applications Security Console. Security Console enables you to create and manage roles and selected users.

For more information about using the Security Console, see *Using the Security Console*.

> **Note:** In previous releases, Oracle Integration Cloud (OIC) Generation 2 handled workflow and fee processing. Beginning with Release 24B this has changed. OCI Process Automation now handles workflow and fee processing. For the time being the delivered proxy user role name still contains *OIC.*

To create the proxy user:

1. Navigate to the Security Console.

   To navigate to the Security Console, you have these options:
   - Click Setup and Maintenance on the Agency Springboard, and in Functional Setup Manager, click the task: *Create OCI Process Automation Proxy User* in the Permit Types functional area.
   - Click Setup and Maintenance on the Agency Springboard, and on the Fusion Applications home page, select **Navigator** > **Tools** > **Security Console.**
2. Click the Users tab.
3. On the User Accounts page, click **Add User Account.**
4. On the Add User Account page in the User Information section, enter a **Last Name** and **User Name** of your choice.

   The name given for the proxy user will be displayed on the Status History page of the transaction detail pages, for permits, planning applications, and so on. As such, you may want to use a generic name, such as *System, Workflow,* or something similar.
5. Enter a **Password** of your choice and confirm it.
6. Click **Add Role** for the Roles grid, and assign this role to your proxy user:
   - Role Name: *PSCR Proxy User for OIC*
   - Role Code: *CUSTOM_PSCR_OIC_PROXY_USER*
7. Click **Save and Close.**

8. Add the proxy user information to the connectors for your workflow process definitions.

   For each connector, click the Security icon on the right toolbar to add the proxy user information.

   > **Note:** Using the *OAuth* or *Global Credential* security type is recommended. You set up the proxy user to use the OAuth and Global Credential security type in the OCI Process Automation administrative Workspace area, where you select *Credentials* from the Navigator menu.

   For more information on managing security credentials in Workspace, see *Manage Credentials in Workspace*.

Setting Up OCI Process Automation Integration

This topic describes how to set up the integration between Oracle Permitting and Licensing and Oracle Cloud Infrastructure Automation Integration.

Before you begin your Oracle Permitting and Licensing implementation, you need to configure the connection between OCI Process Automation and Oracle Permitting and Licensing. OCI Process Automation provides the capability to design and process:

- Workflow for your transactions.
- Decision models for calculating fees.

These steps should be done within the *Initial Set Up* functional category for your offering in the Functional Setup Manager. The process involves setting up the users, groups (roles), and authentication in Oracle Identification Cloud Service (IDCS) required for OCI Process Automation and the Permitting and Licensing system to interact.

> **Note:** You need to complete these tasks on each pod (test, development, production, and so on) for each pairing of Oracle Identity Cloud Service and Oracle Cloud Infrastructure Process Automation.

Create a Trusted User on IDCS

1. Sign in to the Identity Cloud Service console.
2. Select *Identity > Domains* and select your domain.
3. Select *Users*.
4. Click **Create User.**
5. On the Create user page, add these values:
   a. **First name:** *PSCR*
   b. **Last name:** *PROXY_USER*
   c. **Use the email address as the username:** *Deselect*
   d. **Username:** PSCR_PROXY_USER
   e. **Email:** *no-reply@oracle.com*

   > **Note:** The First Name, Last Name, and Email values can be any value. The Username value must be *PSCR_PROXY_USER.*

6. Click **Create.**

Create the PSCR Submitter Group

1. Sign in to the Identity Cloud Service console.
2. Select *Identity > Domains* and select your domain.
3. Select *Groups*.
4. Click **Create Group.**
5. On the Create group page, add these values:
    a. **Name:** *PSCR Submitter Group*
    b. **Users grid:** select the trusted user you just created.
6. Click **Create.**

Set Up Cloud Service Application Roles

1. In Functional Setup Manager, complete the *Run User and Roles Synchronization Process* task in the Initial Setup Functional area.
2. In IDCS import users into the identity domain.
3. Sign in to the Identity Cloud Service console.
4. Select *Identity > Domains* and select your domain.
5. Select Oracle Cloud Services.
6. Open the IDCS app named *Process Automation Service.*
7. Select Application roles under Resources on the left.
8. Expand the *ServiceAdministrator* role and click Manage for Assigned groups.
9. On the Manage group assignments page, click the plus sign in the Show available groups link beneath the Assigned groups grid to expose the Available groups grid.
10. In the Available groups grid, search for and select the following groups individually, and click **Assign.**
    a. *PSCR Submitter Group*
    b. *PSC System Administrator*
    c. *PSC Business Analyst*
    d. *PSC Custom Manage All Workflow Tasks*
    e. *PSC Custom Administer Workflow*
11. Click **Close.**
12. In the Application roles list, expand the *ServiceBusinessUser* role and click **Manage** for Assigned groups.
13. On the Manage group assignments page, search for and select the *PSC Agency Staff* group and click **Assign.**

Confirm OCI Process Automation Authentication

1. Access the OCI Process Automation designer URL.

    It will look similar to:

    https://opa-xxx-xxx-xxxxxxx.process.oci.oraclecloud.com/process/designer

    It is recommended to create a bookmark for easy access.

    You can get the base URL by accessing the Primary Audience URL, as displayed in the IDCS App for the Process Automation Service.

    Assuming you are still on the Manage group assignments page from the previous step, you can also access this URL by clicking OAuth configuration under Resources on the left. In the Configure application APIs that need to be OAuth protected, locate the **Primary audience** URL.

> **Note:** Keep the Primary audience URL available as you'll need it in the next task..

2. Enter your user ID and password and confirm you can sign in.
   Use a Fusion Application user ID assigned at least to the *PSC System Administrator* role.

Create OAuth Credentials

1. Sign in directly to the individual identity domain.
   By viewing the My Profile menu, you can see if you are signed into a specific domain or through the (default) domain.
   If you are logged into the default domain, you an get the URL for an individual identity domain, by selecting Identity > Domains. In the domains grid select the domain you are currently configuring. On the Over view page for that domain click Copy for the **Domain URL** field.
   Open a new browser window and copy that URL into the search bar, adding */ui/vi/adminconsole* to the URL.
2. Open your REST client application, such as Postman, Curl, or similar.
3. Add the Domain URL value to the Identity Cloud Service console and select the POST action.
   Add */admin/v1/Apps* to your base Domain URL.
   The URL will look similar to:
   https://idcs-abc123xxxxxxxxxx.identity.oraclecloud.com/admin/v1/Apps
4. In the body, copy and paste the following JSON, updating the *name* and the *redirectUris* attributes.
   The user assigned to *name* should be created in Permitting and Licensing with sufficient privileges to call Oracle Permitting and Licensing from OCI Process Automation through REST APIs. This isn't the user created previously in the step where you created a trusted user on IDCS. This user will be added to your workflow process connectors.
   The *redirectUris* URL should use the URL displayed as the Primary audience URL for your domain to which you add */icsapis/agent/oauth/callback*.

```
{
 "schemas": [
 "urn:ietf:params:scim:schemas:oracle:idcs:App"
 ],
 "displayName": "OPA App for PSCR OAuth Inbound",
 "isOAuthClient" : true,
 "description": "OPA App for PSCR OAuth Inbound",
 "active": true,
 "clientType": "confidential",
 "name": "xxxxx",
 "basedOnTemplate": {
 "value": "CustomWebAppTemplateId"
 },
 "redirectUris": [
 "<OCI Process Automation Base URL>/icsapis/agent/oauth/callback"
 ],
 "logoutUri": "",
 "postLogoutRedirectUris": [""],
 "allUrlSchemesAllowed": true,
 "allowedGrants": [
 "client_credentials",
 "password",
```

```
 "urn:ietf:params:oauth:grant-type:jwt-bearer",
 "authorization_code",
 "refresh_token"
 ]
}
```

5. Retrieve and copy your OAuth token.
    a. In Oracle Identity Service console, select the Profile menu in the upper right of the header.
    b. Select *My profile*.
    c. Under Resources, click My access tokens.
    d. Under My access tokens, click the **Select app role** field.
    e. Select *Identity Domain Administrator.*
    f. Consider the **Token expires in mins** field. You may want to set this to a higher value than the default to provide additional time. Keep in mind that if you need to retry this operation for any reason, your token may have expired, so you'd need to recreate the token.
    g. Click **Download token.**
    h. Open the downloaded token in a text editor, such as Notepad.
    i. Copy the contents of the token file.
6. Return to your REST client, click the Authorization tab, select *Bearer Token* for the **Type** field, and copy the token file contents into the **Token** field.
7. Click **Send.**
8. Return to the Oracle Identity Service console and open your domain.

    In the navigation breadcrumbs at the top left, click Identity, then click Domains under Identity on the left, click on your domain in the Domains in... grid.
9. Under Identity domain, select Integrated applications.
10. On the Integrated applications page, select *OPA App for PSCR OAuth Inbound*.
11. Under OAuth configuration, click **Edit OAuth configuration.**
12. On the Edit OAuth configuration page:
    a. Select Add resources.
    b. Under Resources click **Add scope.**
    c. On the Add scope page, select *Oracle Applications Cloud (Fusion).*
    d. Expand the Oracle *Applications Cloud (Fusion)* row, and select the scope that appears.

        > **Note:** The string reflects that your Fusion Application instance is a consumer of all Fusion Application resources.

    e. Click **Add** and **Save changes.**

Set OAuth Credentials in OCI Process Automation

1. Return to OCI Process Automation designer.

    For example:

    https://opa-xxx-xxx-xxxxxxx.process.oci.oraclecloud.com/process/designer
2. Select the Workspace node in the left navigation column.
3. In the Workspace, select Credentials.
4. In the upper right click **Create global credentials,** and select *OAuth credentials.*
5. On the Add new OAuth credential page, add these values:

a. **Credential Name:** *OPAL_OPA_GLOBAL_OAUTH*
b. **Target URL:** Add the Fusion Application base URL for the current pod, such as *https//fa-xxxx-xx-xx.fa.xx.oraclecloud.com.* This is the base URL when signing on to Oracle Permitting and Licensing.
c. **Client Id:** Add the name you added to the JSON in a previous step when creating the OAuth credentials.
d. **Client Secret:** Return to your OAuth configuration in the Oracle Identity Service console for the current domain, and under General Information, click **Show secret.**

   From the Client secret pop-up window, copy the secret, and paste it into the **Client Secret** field.
e. **Scope:** Return to your OAuth configuration in the Oracle Identity Service console for the current domain, and under Token issuance policy, select and copy the **Scope** value for the Oracle Applications Cloud (Fusion) resource. Paste it into the **Scope** field.
f. **OAuth Token URL:** Select *Local Identity Domain.*
g. **Description:** Add a description, such as, *Global OAuth credentials for callbacks to <your Permitting and Licensing pod>.*
h. Click **Submit.**

3 Setting Up Structured Workflow

Workflow Basics

You define your transaction workflow using the Processes feature of OCI Process Automation.

You can incorporate workflow processing for all Oracle Permitting and Licensing offerings, but structured workflow processes apply specifically to these offerings:

- Permits
- Planning and Zoning
- Business Licenses

For Permits, Planning and Zoning, and Business Licenses, you define structured processes to run your workflow. For more information on structured processes, see *Develop Structured Processes*.

Code Enforcement implements workflow using a different process definition type, which is covered in a separate chapter. See *Dynamic Workflow Overview* for more information on Code Enforcement workflow.

> **Note:** Oracle provides a Solution Package with sample workflow configurations. It is highly recommended that you clone these samples and use them as starting points to create your own workflow.

> **Note:** Currently, in the context of data object parameters, data association parameters, and REST resource attributes, the terms *record, transaction, permit, planning application,* and *business license* are interchangeable.

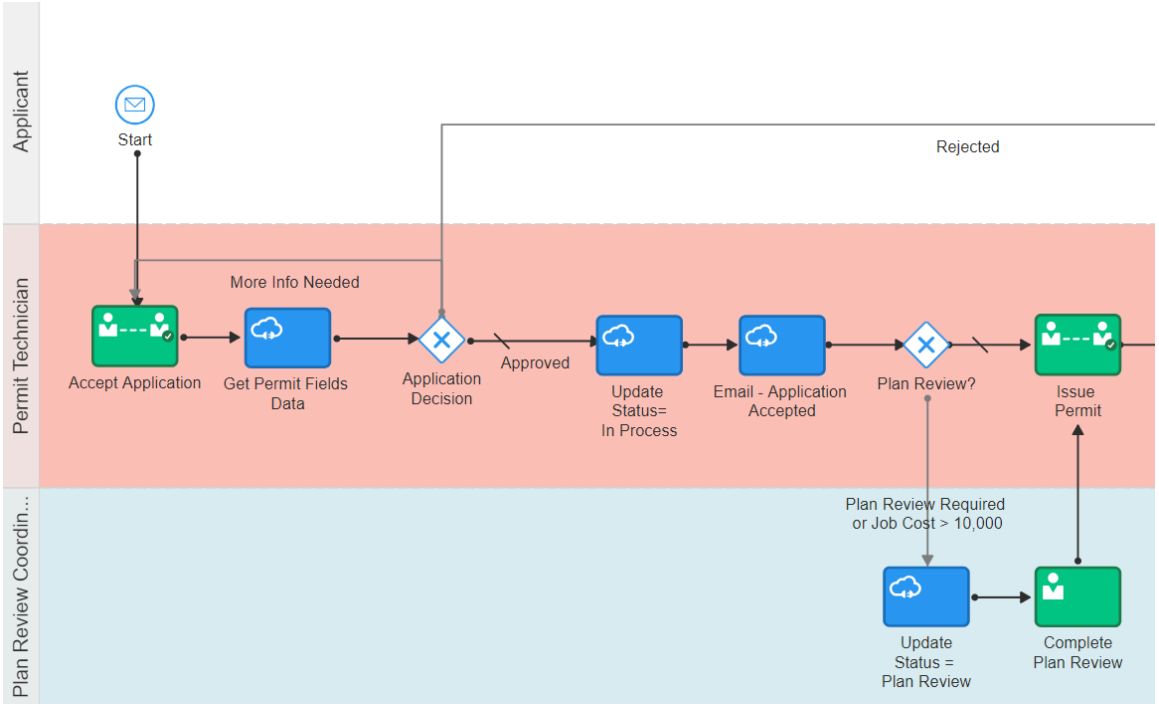
Reviewing a Sample Process Definition

A process definition provides a defined flow for processes such as the transaction lifecycle of a permit. This flow can include system tasks, human tasks, and decision gateways. You define your flow using the Process feature in OCI Process Automation. The Process feature provides a visual design environment to help you create easily understood workflow process definitions.

> **Note:** Currently, in the context of data object parameters, data association parameters, and REST resource attributes, the terms *record, transaction,* and *permit* are interchangeable.

Let's look a a sample process definition for a building permit.

This image shows the first half of the sample process, from the time the permit is submitted until it is issued.



The following table identifies the types of objects shown in the illustration:

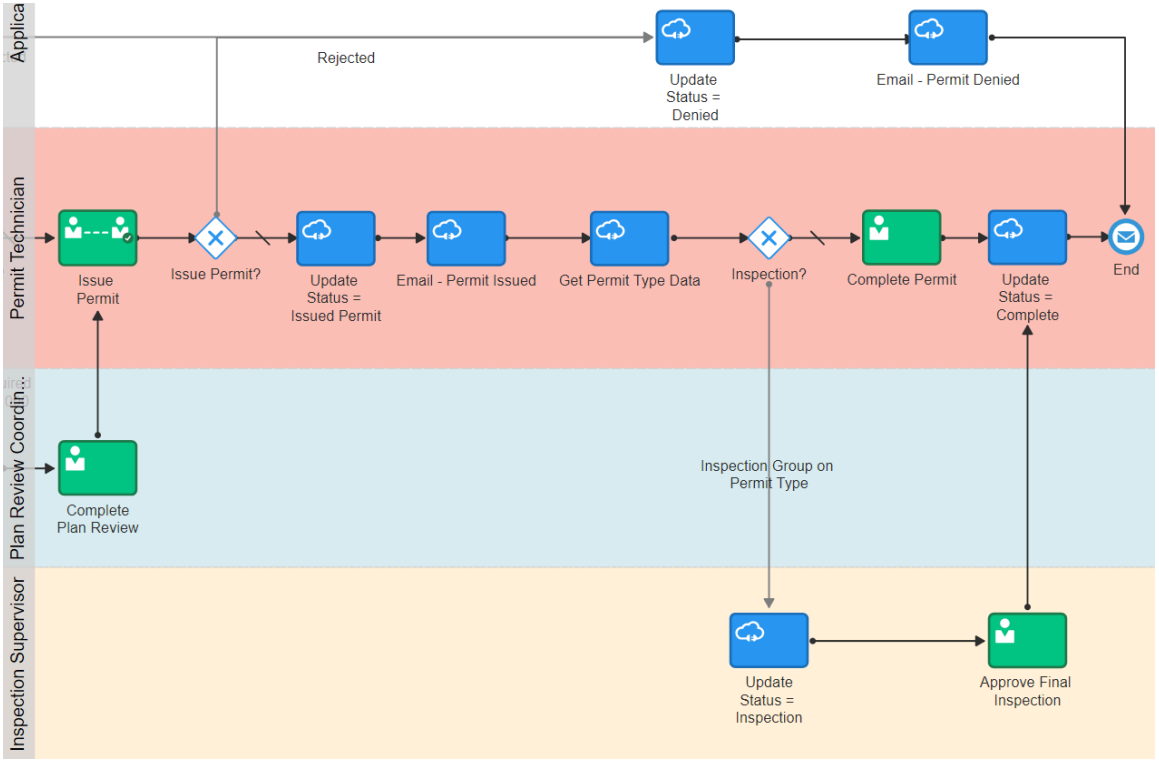
Object	Description
Swimlanes	Horizontal bands in the process map represent the roles involved in the process.
Start and End Events	All paths through the workflow process must begin at the Start event and finish at the End event.
Human tasks	Green boxes with an image of a person represent tasks that are performed by humans.
System tasks	Blue boxes with an image of a cloud represent tasks that the system performs.
Gateways	White diamonds represent decision points, where the process flow can branch based on criteria you define.
Arrows	One-directional arrows define flows through the process. Gateways are the only objects that have multiple exit arrows. The exit arrow with a slash through it represents the default option after a gateway. All other exit arrows contain business logic for defining the conditions when the arrow is used.

With these definitions in mind, let's look at the sample process flow:

1. **Start:** The process starts when a permit application is submitted, which sends a message to OCI Process Automation to instantiate the workflow process.
2. **Accept Application:** A human performs the task of accepting the application and selecting a task status that represents the task outcome.

3. **Get Permit Fields Data:** This system task retrieves permit field data to be used later in the process, when it's time to determine whether a plan review is required.
4. **Application Decision:** Exit arrows from this gateway determine the next step based on the outcome of the Accept Application human task.
    a. If more information is needed, the application acceptance task is reinstantiated. This loop continues until the task has a different outcome.
    b. If the application is rejected, a system task updates the permit status to *Denied,* then another system task sends the applicant an email notification that the permit was denied, then the process ends.
    c. If the outcome is anything else, the process continues.
5. **Update Status = In Process:** This system task updates the permit status to *In Process.*
6. **Email - Application Accepted:** This system task notifies the applicant that the permit was accepted.
7. **Plan Review:** Exit arrows from this gateway determine the next step based on the outcome of the Accept Application task and based on the job cost that was retrieved by the Get Permit Fields Data task:
    a. If the Accept Application outcome indicates that a plan review is required, or if the job cost is greater than 10,000, the **Update Status = Plan Review** system task updates the permit status to *Plan Review,* then a human completes the **Complete Plan Review** human task. When the Complete Plan Review is complete, the process continues.
    b. If a plan review is not required, the process continues.
8. **Issue Permit:** A human performs the task of issuing the permit and enters a task status that represents the task outcome (whether the permit was issued or rejected).

The following image shows the remainder of the sample workflow, after a human completes the Issue Permit task.



These steps describe the remainder of the workflow process, after the human task for issuing a permit:

1. **Issue Permit:** Exit arrows from this gateway determine the next step based on the outcome of the task for issuing a permit:
    a. If the permit is rejected, the **Update Status = Denied** This system task updates the permit status to *Denied,* then the **Email - Permit Denied** system task notifies the applicant that the permit was denied, then the process ends.
    b. If the outcome is anything else, the process continues.
2. **Update Status = Issued Permit:** This system task updates the permit status to *Issued Permit.*
3. **Email - Permit Issued:** This system task notifies the applicant that the permit was issued.
4. **Get Permit Type Data:** This system task retrieves permit type information for use in determining whether an inspection is needed.
5. **Inspection:** Exit arrows from this gateway determine whether an inspection is needed:
    a. If the permit type includes an inspection group, the **Update Status = Inspection** system task updates the permit status to *Inspection.* A human then completes the **Approve Final Inspection** task and enters the task outcome. The process then continues.
    b. If an inspection is not required, a human performs the **Complete Permit** task and enters the task outcome. The process then continues.
6. **Update Status = Complete:** this system task updates the permit status to *Complete.*
7. The process ends.

Setting Up the Communications Connector

The communications connector enables OCI Process Automation to send data to the communications center in the Oracle Permitting and Licensing system using a POST operation. This connector is used when a workflow process definition includes a communication task, such as sending a permit applicant an email when the permit status changes.

> **Note:** Oracle provides a Solution Package with sample process definitions with preconfigured connector configurations. It is recommended to clone these samples and use them as starting points for your own process definitions. The instructions in this procedure explain how to set up the communications connector from scratch and should be considered only a sample for illustration purposes.

The following procedure explains how to set up the communications connector with the specific integration information that is required by the Public Sector system.

> **Note:** Currently, in the context of data object parameters, data association parameters, and REST resource attributes, the terms *record, transaction,, permit, business license, or planning application* are interchangeable.

To set up the communication connector:

1. (Optional) If you are building the process application from scratch, you can download sample JSON from My Oracle Support.

    Go to My Oracle Support, access Doc ID 2449735.1, *Public Sector Compliance and Regulation: JSON Files for Transaction Integration,* and download the following files that you will use later in this procedure:

    - RequestCommunications.json

- ResponseCommunications.json

2. Access the main console in OCI Process Automation.
3. In the list of applications, click the process application with your transaction workflow.
4. Click the **Connectors** node at the top.
5. Click the **Add** button, then in Add component drawer, expand **Connectors** > **REST API**
6. In the Add component drawer, enter the following:

Page Element	Description
Title	Enter a descriptive name such as *CommunicationsConnector.* **Note:** The name *CommunicationsConnector* is suggested, however, you can choose your own name if needed. The title is the user friendly term.
Identifier Name	Automatically created by the title you enter but can be changed before saving. This is the internal system ID for the connector.
Base URL	Enter the URL for your Oracle Permitting and Licensing REST API resources. The URL follows this pattern, where *ServerName* is the server name for your instance of the application: https://*ServerName*/fscmRestApi/resources/11.13.18.05

7. Click **Create.**
8. Click the **Security** icon on the right.
9. On the Security drawer, for **Security Type** select either *OAuth* or *Global Credential,* the recommend options.

   This is where you provide the proxy user information. See *Setting Up OCI Process Automation Proxy User*.
10. Depending on the security type selected, enter the appropriate information.
11. Click **Save**.
12. Click **Add** in the header.
13. Expand the new Resource section that appears, and enter the following values:

Field	Value
Name	*OutboundCommunications*
Path	*publicSectorCommunicationRequests* When added to the base URL, this completes the path to the communications-related REST APIs.

Field	Value
	Note: If you need to call a Business Rules Framework event to run a business rule to send a notification only if the scenario meets your criteria, you can set up the connector to call the Business Rules Framework Request API (publicSectorBusinessRulesFrameworkRequests).

14. In the Operations section, click the **Add** button and then select *POST* from the drop-down menu.
15. Click the new POST operation.
16. Enter *Trigger transaction communications* in the **Description** field.
    You can leave the default values in the other fields, including leaving the **Path** field blank.
17. Click **Request**
18. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
19. Enter *RequestCommunications* in the **Name** field.
20. Click **Schema.**
21.  Click the **Import from File** icon next to the **Schema** button.
22. Locate and upload the *RequestCommunications.json* file that you downloaded from My Oracle Support.
    The imported JSON code appears in the Import Business Object from JSON window.
23. Click the **Import** button at the bottom of the window to save the code and close the window.
24. Ensure that the following values now appear for the POST operation request:

Page Element	Value
Body	*BusinessData.RequestCommunications*
Media Type	*Custom*
Media Type details	*application/vnd.oracle.adf.resourceitem+json*

25. Click **Response.**
26. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
27. Enter *ResponseCommunications* in the **Name** field.
28. Click **Schema.**
29.  Click the **Import from File** icon next to the **Schema** button.
30. Locate and upload the *ResponseCommunications.json* file that you downloaded from My Oracle Support.
    The imported JSON code appears in the Import Business Object from JSON window.
31. Click the **Import** button at the bottom of the window to save the code and close the window.
32. Ensure that the following values appear for the POST operation response:

Page Element	Value to Enter
Body	*BusinessData.ResponseCommunications*

Page Element	Value to Enter
Media Type	*application/JSON*

33. Click **Apply.**
34. Click **Save.**

Setting Up the Transactions Connector

The transactions connector enables OCI Process Automation to exchange transaction-related information with the Oracle Permitting and Licensing system.

Note: Oracle provides a Solution Package with sample process definitions with preconfigured connector configurations. It is recommended to clone these samples and use them as starting points for your own process definitions. The instructions in this procedure explain how to set up the communications connector from scratch and should be considered only a sample for illustration purposes.

Note: Currently, in the context of data object parameters, data association parameters, and REST resource attributes, the terms *record, transaction, permit, business license, or planning application* are interchangeable. In some resource attributes, such as paths or parameters, *trans* is used in place of *transaction* for simplicity.

Procedure Overview for Setting Up the Transactions Connector

Setting up the Transactions Connector comprises multiple procedures with multiple steps. The following set of steps outline the high-level set of procedures involved with this task. Each item in the following list is explained in more detail in the following sections, in the listed sequence.

1. Download JSON Files for Workflow Integration Configuration (only if you are building from scratch and not using the provided samples in the Solution Package).
2. Create the Transactions Connector
3. Add the Transaction Resource
4. Add the PATCH Operation for Transaction Statuses
5. Add the GET Operation for Transaction Base Data
6. Add the GET Operation for Transaction Fields Data
7. Add the GET Operation for Transaction Data
8. Add the GET Operation for Transaction Assignee
9. Add the GET Operation for Transaction Type Resource
10. Add the Transaction Type Data
11. Add the POST Operation for the Business Rules Request.

Step 1: Download Required JSON Files for Workflow Integration Configuration

For creating integrations from scratch, Oracle provides a set of JSON files for defining the integration between OCI Process Automation and the Oracle Permitting and Licensing system. Download all of the files first so that you can access them easily while completing the procedures documented in this topic.

To download the required JSON:

1. Sign on to My Oracle Support.
2. Access Doc ID 2449735.1, *Public Sector Compliance and Regulation: JSON Files for Transaction Integration.*
3. Download the following files to a local folder:
    - RequestTransStatusUpdate.json
    - ResponseTransStatusUpdate.json
    - ResponseTransBase.json
    - ResponseTransFields.json
    - ResponseTransData.json
    - ResponseTransAssignee.json
    - ResponseTransType.json

Step 2: Creating the Transaction Connector

> **Note:** This procedure explains how to create the transaction connector. Additional procedures that follow this one explain how to set up the operations for this connector.

To set up the transactions connector:

1. Access the main console in OCI Process Automation.
2. In the list of applications, click the application for your workflow process.
3. Select the **Connectors** node in the toolbar.
4. Click the **Create** button, then in the pop-up menu under the Create button, select **Connectors** > **REST API**
5. In the Add component drawer, enter the following:

Page Element	Description
Name	Enter a descriptive name, such as *TransactionConnector.* **Note:** The name *TransactionConnector* is suggested, however, you can choose your own name if needed. This documentation refers to *TransactionConnector.*
Identifier Name	Automatically created by the title you enter but can be changed before saving. This is the internal system ID for the connector.

Page Element	Description
Base URL	Enter the URL for your Oracle Public Sector Cloud REST API resources. The URL follows this pattern, where *ServerName* is the server name for your instance of the application: https://*ServerName*/fscmRestApi/resources/11.13.18.05

6. Click **Create.**
7. Click the **Security** icon from the toolbar on the right.
8. On the Security drawer, for **Security Type** select either *OAuth* or *Global Credential,* the recommend options. This is where you provide the proxy user information. See *Setting Up OCI Process Automation Proxy User*.
9. Depending on the security type selected, enter the appropriate information.
10. Click **Save**

Step 3: Add the Transactions Resource

Note: Before starting this procedure, be sure to complete the procedure "Setting Up the Transactions Connector."

1. Access the main console in OCI Process Automation.
2. In the list of applications, click the application for your workflow process.
3. Click the **Connectors** option.
4. Click the **TransactionConnector** connector.
5. In the Resources section of the Rest Connector Editor, click **Add.**
6. Expand the new Resource section that appears, and enter *TransactionResource* in the **Name** field.

Step 4: Add the PATCH Operation for Transaction Statuses

Workflow in OCI Process Automation uses the PATCH operation to update the status of a transaction.

To set up the PATCH operation:

1. In the **Operations** section of TransactionResource, click the **Add** button and then select **PATCH operation** from the drop-down menu.
2. Click the new **PATCH** operation.
3. Enter the following information:

Page Element	Value
Name	*patchTransactionStatus*
Path	*{transResource}/{transRecordKey}* Although you can choose different names for the resource name and record key parameters, this procedure assumes that you use the given values.
Documentation	*Update transaction status.*

4. Click **Request**
5. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
6. Enter *RequestTransStatusUpdate* in the **Name** field.
7. Click **Schema.**
8.  Click the **Import from File** icon next to the **Schema** button.
9. Locate and upload the *RequestTransStatusUpdate.json* file that you downloaded from My Oracle Support.

   The imported JSON code appears in the Import Business Object from JSON window.
10. Click the **Import** button at the bottom of the window to save the code and close the window.
11. Ensure that the following values now appear for the PATCH operation request:

Page Element	Value to Enter
Body	*BusinessData.RequestTransStatusUpdate*
Media Type	*Custom*
Media Type details	*application/vnd.oracle.adf.resourceitem+json*

12. In each row of the **Parameters** list, click the *Enter a description* text and enter a description.

    These are example descriptions:

Parameter	Description
transResource	*Transaction Resource Name*
transRecordKey	*Transaction Record Key*

13. Click **Response.**
14. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
15. Enter *ResponseTransStatusUpdate* in the **Name** field.
16. Click **Schema.**
17.  Click the **Import from File** icon next to the **Schema** button.
18. Locate and upload the *ResponseTransStatusUpdate.json* file that you downloaded from My Oracle Support.

    The imported JSON code appears in the Import Business Object from JSON window.
19. Click the **Import** button at the bottom of the window to save the code and close the window.
20. Ensure that the following values appear for the PATCH operation response:

Field	Value
Body	*BusinessData.ResponseTransStatusUpdate*

Field	Value
Media Type	*application/JSON*

21. Click **Apply.**
22. Click **Save.**

Step 5: Add the GET Operation for Transaction Base Data

> **Note:** Before starting this procedure, be sure to complete the procedure "Adding a PATCH Operation for Transaction Statuses."

The *getTransactionBaseData* operation gets general transaction data that is found in all transactions, such as the permit type, the permit status, and the permit applicant for a permit transaction.

1. Expand the **TransactionResource** resource.
2. In the **Operations** section, click the **Add** button and then select **GET operation** from the drop-down menu.
3. Click the new **GET** operation.
4. Enter the following information:

Field	Value
Name	*getTransactionBaseData*
Path	*{transResource}/{transRecordKey}* Although you can choose different names for the resource name and record key parameters, this procedure assumes that you use the given values.
Description	*Get base transaction data, such as applicant information*

5. Click **Request**
6. In each row of the **Parameters** list, click the *Enter a description* text and enter a description.

   These are example descriptions:

Parameter	Description
transResource	*Transaction Resource Name*
transRecordKey	*Transaction Record Key*

7. Click **Response.**
8. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
9. Enter *ResponseTransBase* in the **Name** field.
10. Click **Schema.**
11.  Click the **Import from File** icon next to the **Schema** button.
12. Locate and upload the *ResponseTransBase.json* file that you downloaded from My Oracle Support.

    The imported JSON code appears in the Import Business Object from JSON window.
13. Click the **Import** button at the bottom of the window to save the code and close the window.
14. Ensure that the following values appear for the GET operation response:

Page Element	Value to Enter
Body	*BusinessData.ResponseTransBase*
Media Type	*application/JSON*

15. Click **Apply.**

Step 6: Add the GET Operation for Transaction Fields Data

The *getTransactionFieldsData* gets field data from the application intake form configured using the Intake Form Designer.

1. In the **Operations** section of the Transactions Resource, click the **Add** button and then select **GET operation** from the drop-down menu.
2. Click the new **GET** operation.

   The new GET operation has the default name of *GetTransactionResources.*
3. Enter the following information:

Field	Value
Name	*getTransactionFieldsData*
Path	*{transResource}/{transRecordKey}/child/FieldGroups* Although you can choose different names for the resource name and record key parameters, this procedure assumes that you use the given values.
Description	*Get specific transaction data, such as job cost.*

4. Click **Request.**
5. In each row of the **Parameters** list, click the *Enter a description* text and enter a description.

   These are example descriptions:

Parameter	Description
transResource	*Transaction Resource Name*
transRecordKey	*Transaction Record Key*

6. Click **Response.**
7. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
8. Enter *ResponseTransFields* in the **Name** field.
9. Click **Schema.**
10.  Click the **Import from File** icon next to the **Schema** button.
11. Locate and upload the *ResponseTransFields.json* file that you downloaded from My Oracle Support.

    The imported JSON code appears in the Import Business Object from JSON window.
12. Click the **Import** button at the bottom of the window to save the code and close the window.
13. Ensure that the following values appear for the GET operation response:

Page Element	Value to Enter
Body	*BusinessData.ResponseTransFields*
Media Type	*application/JSON*

14. Click **Apply.**
15. Click **Save.**

Step 7: Add the GET Operation for Transaction Data

The *getTransactionData* combines getTransactionBaseData and getTransactionFieldsData into a single operation, which you can use instead of using getTransactionBaseData and getTransactionFieldsData separately.

1. In the **Operations** section, click the **Add** button and then select **GET operation** from the drop-down menu.
2. Click the new **GET** operation.

   The new GET operation has the default name of *GetTransactionResources.*
3. Enter the following information:

Field	Value
Name	*getTransactionData*
Path	*{transResource}/{transRecordKey}*

Field	Value
	Although you can choose different names for the resource name and record key parameters, this procedure assumes that you use the given values.
Description	*Combines getTransactionBaseData and getTransactionFieldsData into a single operation.*

4. Click **Request.**
5. In each row of the **Parameters** list, click the *Enter a description* text and enter a description.
   These are example descriptions:

Parameter	Description
transResource	*Transaction Resource Name*
transRecordKey	*Transaction Record Key*

6. Click **Response.**
7. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
8. Enter *ResponseTransactionData* in the **Name** field.
9. Click **Schema.**
10.  Click the **Import from File** icon next to the **Schema** button.
11. Locate and upload the *ResponseTransactionData.json* file that you downloaded from My Oracle Support.
    The imported JSON code appears in the Import Business Object from JSON window.
12. Click the **Import** button at the bottom of the window to save the code and close the window.
13. Ensure that the following values appear for the GET operation response:

Page Element	Value to Enter
Body	*BusinessData.ResponseTransactionData*
Media Type	*application/JSON*

14. Click **Apply.**
15. Click **Save.**

Step 8: Add the GET Operation for Transaction Assignee

The *getTransactionAssignee* operation gets the assigned planner for transactions within the Planning and Zoning offering. You can configure subsequent tasks in the workflow process to reference the retrieved and stored getTransactionAssignee value.

1. In the **Operations** section, click the **Add** button and then select **GET operation** from the drop-down menu.

2. Click the new **GET** operation.

   The new GET operation has the default name of *GetTransactionResources.*
3. Enter the following information:

Field	Value
Name	*getTransactionAssignee*
Path	*publicSectorTransactionLatestAssignees/{transRecordKey}* Although you can choose different names for the resource name and record key parameters, this procedure assumes that you use the given values.
Description	*Get specific transaction data, such as job cost.*

4. Click **Request.**
5. In each row of the **Parameters** list, click the *Enter a description* text and enter a description.

   These are example descriptions:

Parameter	Description
transRecordKey	*Transaction Record Key*

6. Click **Response.**
7. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
8. Enter *ResponseTransactionFields* in the **Name** field.
9. Click **Schema.**
10.  Click the **Import from File** icon next to the **Schema** button.
11. Locate and upload the *ResponseTransAssignee.json* file that you downloaded from My Oracle Support.

    The imported JSON code appears in the Import Business Object from JSON window.
12. Click the **Import** button at the bottom of the window to save the code and close the window.
13. Ensure that the following values appear for the GET operation response:

Page Element	Value to Enter
Body	*BusinessData.ResponseTransAssignee*
Media Type	*application/JSON*

14. Click **Apply.**

15. Click **Save.**

Step 9: Add the Transaction Type Resource

Note: Before starting this procedure, be sure to complete the previous procedures.

1. Access the main console in OCI Process Automation.
2. In the list of OCI Process Automation applications, click the application for your workflow process.
3. Click the **Integrations** option in the left frame.
4. Click the **TransactionsConnector** integration.
5. In the header of the Resources section, click **Add** to create a new transaction type resource.
6. Expand the new Resource section that appears, and enter the following information:

Field	Value
Name	TransactionTypeResource
Path	publicSectorRecordTypes

Step 10: Add the GET Operation for Transaction Type Data

The *GetTransactionTypeData* operation gets data that is associated with the transaction type definition rather than with an individual transaction. For example, this operation can get the overall fee structure for a permit definition, because the fee structure is associated with the permit type.

To set up the GET operations for transaction type data:

1. In the **Operations** section of the Transaction Type Resource, click the **Add** button and then select **GET operation** from the drop-down menu.
2. Click the new **GET** operation.
3. Enter the following information:

Page Element	Value
Name	*getTransactionTypeData*
Path	*{transResource}*
Documentation	*Get transaction type setup data.*

4. Click **Request**
5. In the **Parameters** list, click the *Enter a description* text and enter a description.

   Here is an example description:

Parameter	Description
transResource	*Transaction Resource Name*

6. Click **Response.**
7. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
8. Enter *ResponseTransTypeData* in the **Name** field.
9. Click **Schema.**
10.  Click the **Import from File** icon next to the **Schema** button.
11. Locate and upload the *ResponseTransType.json* file that you downloaded from My Oracle Support.

    The imported JSON code appears in the Import Business Object from JSON window.
12. Click the **Import** button at the bottom of the window to save the code and close the window.
13. Ensure that the following values appear for the GET operation response:

Page Element	Value to Enter
Body	*BusinessData.ResponseTransTypeData*
Media Type	*application/JSON*

14. Click **Apply.**
15. Click **Save.**

Step 11: Add the POST Operation for the Business Rules Request

This step describes how to configure integration with Oracle Permitting and Licensing by calling REST resources directly from workflow process definitions.

Generally, any REST API can be called from your workflow process definition. There are multiple REST API's provided with the Oracle Permitting and Licensing offerings. The sample workflow models provided to get you started include only a handful of the integrations that could be configured, depending on your business requirements.

Each process definition has a set of connectors that you can view on the Integrations tab. In the sample workflow process definitions and documentation the each connector is separated to make it easier to view and describe. However, you may choose to combine all the individual connectors into a single connector.

This procedure outlines how you can configure additional integrations within process definitions to call Oracle Permitting and Licensing REST APIs to incorporate more data to drive your workflow process. This example demonstrates how to configure an integration with a Business Rules Framework API.

For more information on the Business Rules Framework, see *Business Rules Framework Overview*.

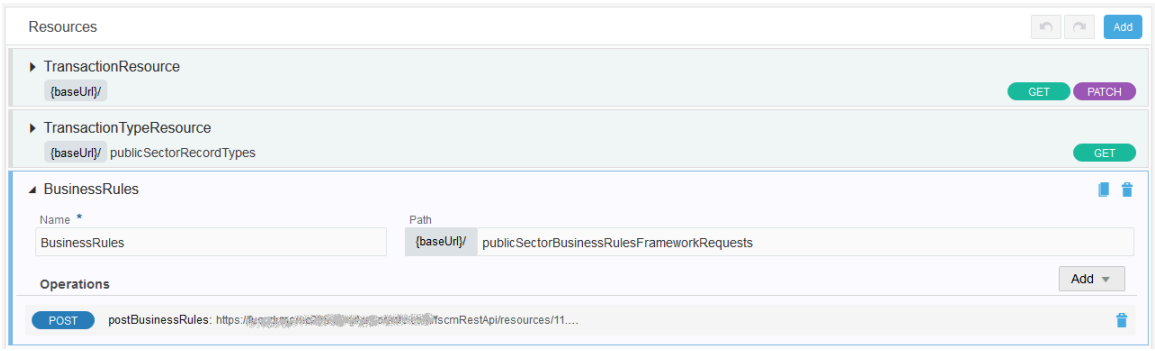
To set up an integration:

1. Open the Transaction Connector.
2. In the Resources box, click **Add.**

3. Expand the added resource and enter these values:

Page Element	Description	Value
Name	The name of the integration or connector. You can enter a custom name as needed.	*BusinessRules*
Path	The name of the REST resource you intend to call.	*publicSectorBusinessRulesFrameworkRequests*

This example illustrates adding a Business Rules resource.

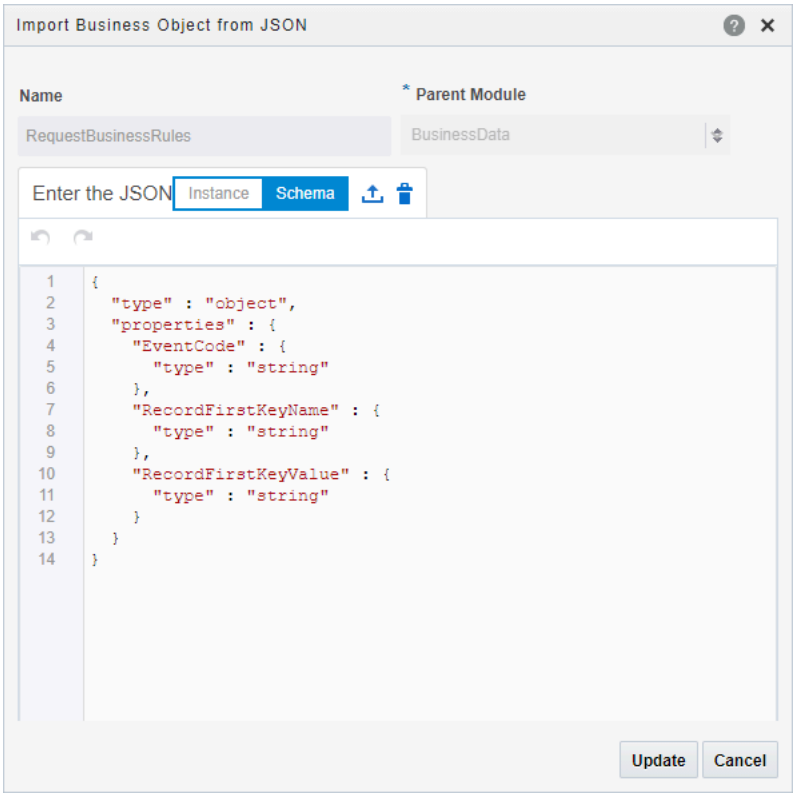


4. On the new resource, click **Add**, and select the appropriate operation.

   In this case, select *POST operation.*
5. Click the newly added operation.
6. (Optional) Add a different name or additional path.
7. Configure the JSON Request.
   a. In the Response and Request section, select the Request tab, and click **Create business object** under the **Body** field to open the Import Business Object from JSON dialog box.
   b. For Name enter a recognizable name for the request, such as *RequestBusinessRules.*
   c. In the Enter the JSON edit box, select **Schema**.
   d. Copy and paste the following JSON.

```
{
 "type" : "object",
 "properties" : {
 "EventCode" : {
 "type" : "string"
 },
 "RecordFirstKeyName" : {
 "type" : "string"
 },
 "RecordFirstKeyValue" : {
 "type" : "string"
 }
 }
}
```

   This example illustrates the request JSON.



e. Click **Import.**
f. Close the Import Business Object from JSON dialog box.

8. Configure the JSON Response.

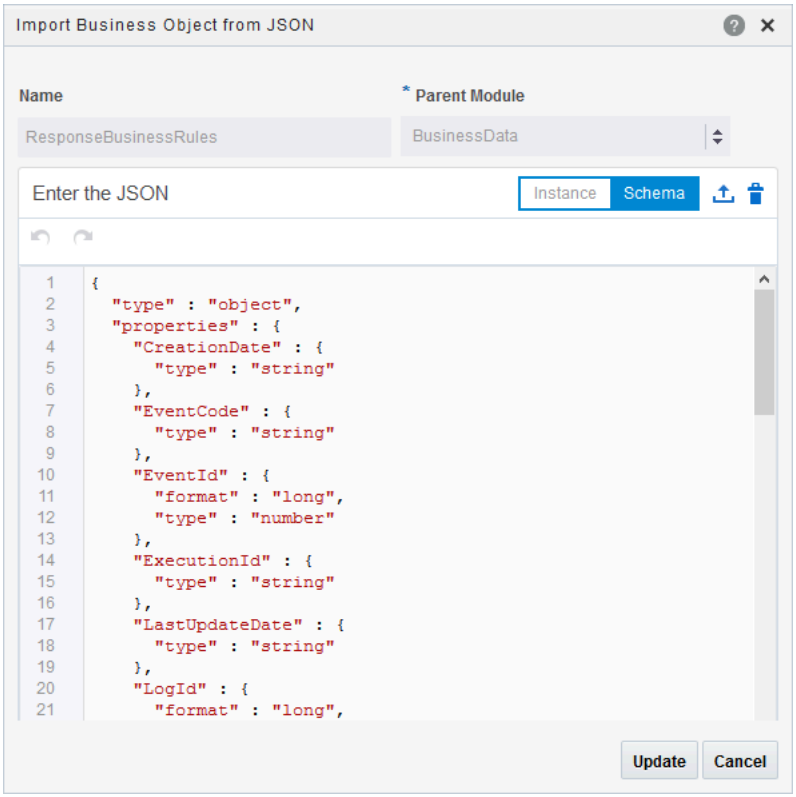
a. In the Response and Request section, select the Request tab, and click **Create business object** under the **Body** field to open the Import Business Object from JSON dialog box.
b. For Name enter a recognizable name for the response, such as *ResponseBusinessRules.*
c. In the Enter the JSON edit box, select **Schema**.
d. Copy and paste the following JSON.

```
{
 "type" : "object",
 "properties" : {
 "CreationDate" : {
 "type" : "string"
 },
 "EventCode" : {
 "type" : "string"
 },
 "EventId" : {
 "format" : "long",
 "type" : "number"
 },
 "ExecutionId" : {
 "type" : "string"
 },
 "LastUpdateDate" : {
 "type" : "string"
 },
 "LogId" : {
```

```
"format" : "long",
"type" : "number"
},
"NoRules" : {
"type" : "string"
},
"RecordFifthKeyName" : {
"type" : null
},
"RecordFifthKeyValue" : {
"type" : null
},
"RecordFirstKeyName" : {
"type" : "string"
},
"RecordFirstKeyValue" : {
"type" : "string"
},
"RecordFourthKeyName" : {
"type" : null
},
"RecordFourthKeyValue" : {
"type" : null
},
"RecordIdentifier" : {
"type" : null
},
"RecordSecondKeyName" : {
"type" : null
},
"RecordSecondKeyValue" : {
"type" : null
},
"RecordSubIdentifier" : {
"type" : null
},
"RecordThirdKeyName" : {
"type" : null
},
"RecordThirdKeyValue" : {
"type" : null
},
"ResourceId" : {
"format" : "long",
"type" : "number"
},
"StopProcess" : {
"type" : "string"
}
```

```
}
```

This example illustrates the response JSON.



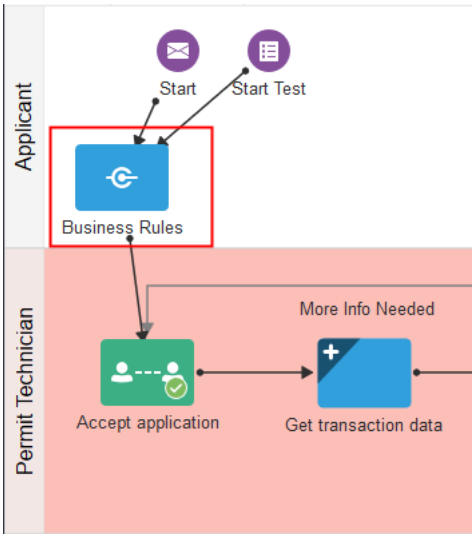
e. Click **Import.**
f. Close the Import Business Object from JSON dialog box.

9. Click **Apply** and **Save**.

10. Add a task to the process definition to incorporate the integration.

    a. Open the process definition.
    b. In the BPMN palette, expand the Integrations section, and drag your integration onto the process definition in the desired location and update the connectors as needed.

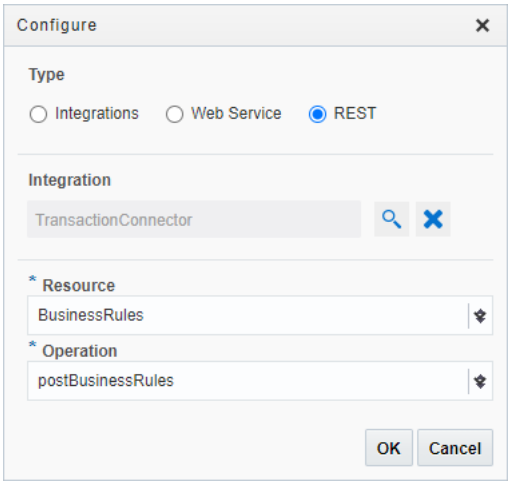
    This example illustrates adding the Business Rules integration to a process definition.



11. Confirm the integration is referencing the correct operation.

    a. Select the task and click Open Properties.
    b. Select Implementation, General and then click **Configure** for the Service Call.

    This example illustrates the integration step referencing the REST operation.

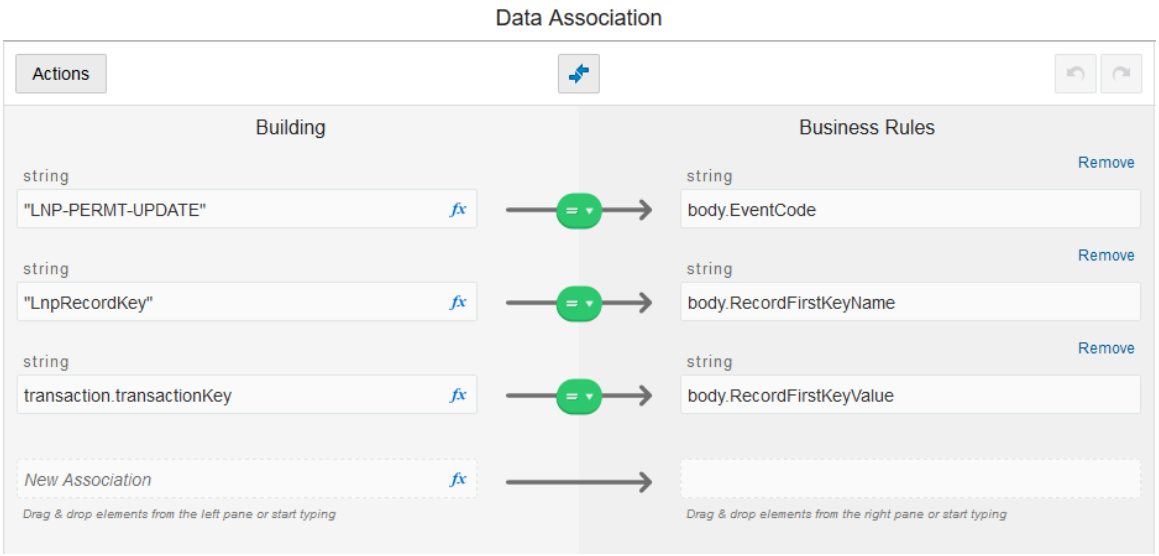


12. Configure data associations.

    a. Select your integration task and select Open Data Association.
    b. On the Data Association page, select the Input tab.
    c. Map the following process definition data attributes to your Business Rules integration attributes, and click **Apply** and **Save.**

Data Object	Integration
"LNP-PERMT-UPDATE" or *"PZ-PLANNING-UPDATE"* Enter these values manually, depending on whether this process definition is being used for Permits or Planning and Zoning.	body.EventCode
"LnpRecordKey" Enter this value manually.	body.RecordFirstKeyName
transaction.transactionKey	body.RecordFirstKeyValue

This example illustrates that data associations between the transaction and the business rules integration.



Setting Up the Sandbox Connector

This topic describes how to configure process definitions to handle requests from transaction types when they are being tested in a development sandbox.

Sandbox Connector Overview

Sandbox connectors enable you to test transaction types, such as permits, planning application types, and business license types, while they are still in draft-mode being designed and tested within the development sandbox. With the Sandbox Connector, you can design an intake form and test it, end-to-end, without needing to publish it. Without the Sandbox Connector, the transaction type intake form will not be able to trigger interaction with the associated workflow process definition.

To complete the integration of the Sandbox Connector, you need to perform these tasks for all existing process definitions:

- Create a Sandbox Connector.
- Update the process definition to use the Sandbox Connector.

Note: This process will need to be completed for all new and existing process definitions.

For more information on sandboxes and testing draft intake forms in the sandbox, see *Working with Sandboxes* and *Testing Intake Forms*.

Creating the Sandbox Connector

To create the Sandbox Connector:

1. Open the process definition.
2. Select **Create** > **External** > **REST**.
3. On the Create REST Connector dialog box, enter these values.

Page Element	Value
Name	*SandboxConnector*
Base URL	*{your host}/fscmRestApi/pscresources/11.13.18.05/dynamic/DynamicRequests*

4. Click **Create.**
5. Click Edit to display the Configuration section.
6. On the General tab, make sure the base URL is correct.
7. Select the Security icon, and enter the proxy user created to access the Oracle Permitting and Licensing data.
   Select either *OAuth* or *Global Credential,* the recommend options. For more information on the proxy user and credentials, see *Setting Up OCI Process Automation Proxy User*.
8. Select the Visibility tab and choose how you want the integration to appear.

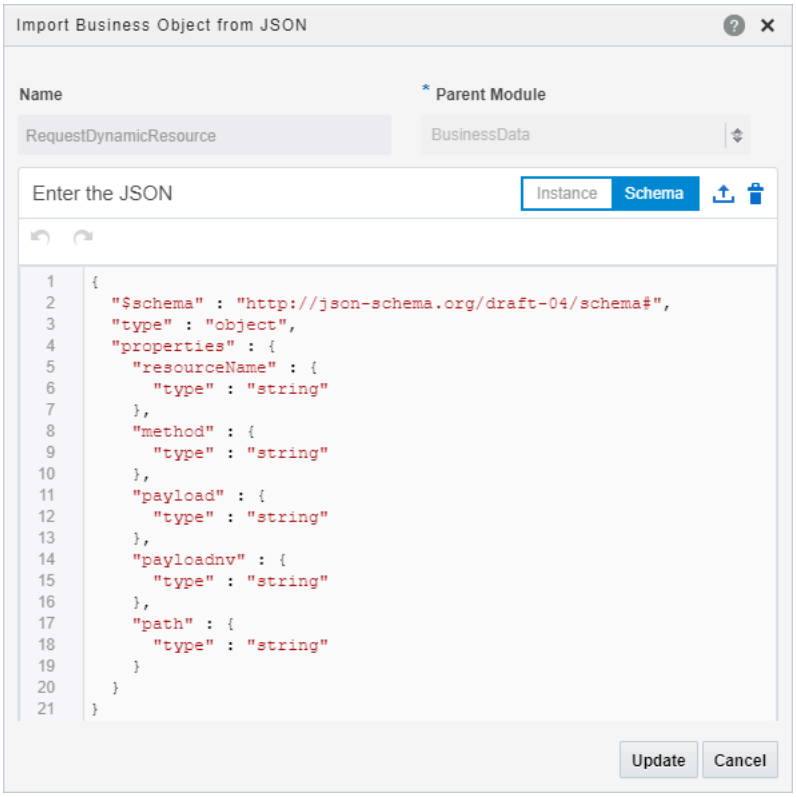
Show operations is recommended.

9. In the Resources box, click **Add.**
10. Enter these values for the resource.

Page Element	Value
Name	*DynamicTransactionResource*
{baseURL}	Blank

11. At the top of the Operations grid, click **Add,** and select *POST operation.*
12. Select the row for the postDynamicTransactionResource operation.
13. In the Resources > Resource box, scroll to the Request/Response area, the Request button will already be selected. Click the plus sign (Create a business object).
14. In the Import Business Object from JSON dialog box, enter these values and click **Import:**

    This example illustrates adding the JSON payload for the Sandbox Connector. Details are in the surrounding text.



Page Element	Value
Name	*RequestDynamicResource*

Page Element	Value

JSON	Select **Schema** and add the following JSON.
	
	`{`
	`"$schema" : "http://json-schema.org/draft-04/schema#",`
	`"type" : "object",`
	`"properties" : {`
	`"resourceName" : {`
	`"type" : "string"`
	`},`
	`"method" : {`
	`"type" : "string"`
	`},`
	`"payload" : {`
	`"type" : "string"`
	`},`
	`"payloadnv" : {`
	`"type" : "string"`
	`},`
	`"path" : {`
	`"type" : "string"`
	`}`
	`}`
	`}`

15. Click the **Response** button, and from the **Body** drop-down list, select *BusinessData.ResponseTransactionData.*
16. Click **Apply** to save your changes.

Update the Process Definition

In this task you will update:

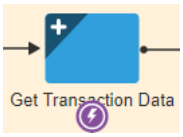
- All calls from the getTransactionData and patchTransactionStatus operations of the Transaction Connector have to include the postDynamicTransactionData operation of the Sandbox Connector.
- All corresponding data associations.

To update getTransactionData calls:

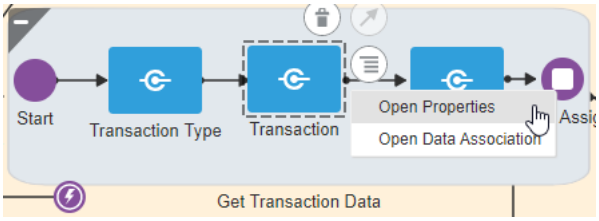
1. Select Processes in the left-hand navigation to display the process.

2. Expand the Get Transaction Data node in the process definition, select the transaction data node, and select Open Properties.

   This example illustrates a collapsed process definition node. Details are in the surrounding text.

   This example illustrates an expanded process definition node. Details are in the surrounding text.

   > **Note:** This node can have other names, such as Get Application, Get Permit Data, and so on. It is typically the first system task in the process definition, where the process definition gets the required initial data, including transaction type, transaction data, and so on.

3. From the **Type** drop-down list, select *Service Call,* and click the Configure button.
4. On the Configure dialog box, click the search icon for the **Integration** field, and select *SandboxConnector.*
5. With *SandboxConnector* displayed in the **Integration** field, update these values, and click **OK**:

Page Element	Value
Resource	*DynamicTransactionResource*
Operation	*postDynamicTransactionResource*

6. Return to the process definition by collapsing the properties pane, select the same node you selected in the steps above, and this time select *Open Data Associations.*

   > **Note:** You may notice warning text referring to undefined variables, which is what you will resolve in the following step.

7. On the Data Association page, make sure the Input tab is selected, and for the field that reads New Association, enter “GET” and then update the data associations accordingly:

   This example illustrates how to map data associations for process definitions. Details are in the surrounding text.

Page Element	Value
transaction.resourceName	*body.resourceName*
transaction.transactionKey	*body.path*
“GET”	*body.method*

8. Select the Output tab and add these values:

Page Element	Value
bodyOutput	*transactionData*

9. Repeat these steps for all calls to the getTransactionData in the process definition.

To update patchTransactionStatus calls:

1. Use same steps above for updating the properties and data associations.
2. Enter these input data associations:

Page Element	Value
transaction.resourceName	*body.resourceName*
transaction.transactionKey	*body.path*

Page Element	Value
"Status:Accepted"	*body.payloadnv*
"PATCH"	*body.method*

3. Enter these output data associations:

Page Element	Value
bodyOutput	transactionData

Use the

Integration JSON Payload Attributes

The integration requires a POST operation request from the OCI Process Automation process definitions.

The payload attributes are:

Attribute	Description
resourceName	Dynamic resource name, such as the permit type.
method	REST method for the dynamic resource, such as GET, POST, PATCH.
path	To access a specific row and child rows of the dynamic resource.
payload	Standard REST payload to pass to POST/PATCH requests. Use either **payload** or **payloadnv**.
payloadnv	Comma delimited key value pairs to pass to the POST/PATCH request. Use either **payload** or **payloadnv**.
sandbox	Optional property if the sandbox context is needed. By default it is true, which means the sandbox exists for the resource. The integration always gives the sandbox context by default unless this property is set to "N".

Setting Up Process Definitions for Workflow

Workflow manages status updates throughout the transaction lifecycle and is an essential part of your setup. This topic provides information for creating your workflow process definitions.

> **Note:** Currently, in the context of data object parameters, data association parameters, and REST resource attributes, the terms *record, transaction,* and *permit, business license, or planning application* are interchangeable. The abbreviation *trans* is often used to represent *transaction.*

Setting Up Data Objects for a Process

Data objects provide a structure for storing data sent from the Oracle Permitting and Licensing system. Every process definition that you create needs the same data objects, including:

- Simple string data definitions to store identifying information about the transaction and transaction type.
- Business object data definitions to store transaction base data, field data, and transaction type data.

  The data definition for fields includes *all possible* fields that can be added to an intake form, even though the transaction may use only a subset of fields that are appropriate for the type of transaction. Any fields that are not part of a specific transaction type remain blank when the workflow process retrieves the field data.

You must set up your data objects before you continue to this topic's additional procedures for defining data associations.

> **Note:** Before you set up your data objects, you need to set up the transaction connector for the application. This is because the transaction connector's GET operations provide the underlying schema for the data. Setting up the transaction connector is described in the topic *Setting Up the Transactions Connector*.

To set up data objects for a process definition:

1. Access the process definition in OCI Process Automation.
2. Click **Data Objects.**
3. Set up the data definition for transaction base data:
   a. In the Data Objects window, click **Add.**
   b. In the Create Process Data Object window, enter the following information:

Page Element	Value
Name	*transactionBaseData*
Data Type	*Business*
The drop-down list for data types	*BusinessData.ResponseTransBase*

c. Click **Create** to create the data definition and return to the Data Objects window.

4. Set up the data definition for transaction field data:
   a. Click **Add.**
   b. Enter the following information:

Page Element	Value
Name	*transactionFieldsData*
Data Type	*Business*
The drop-down list for data types	*BusinessData.ResponseTransFields*

   c. Click **Create.**
5. Set up the data definition for transaction type data:
   a. Click **Add.**
   b. Enter the following information:

Page Element	Value
Name	*transactionTypeData*
Data Type	*Business*
The drop-down list for data types	*BusinessData.ResponseTransType*

   c. Click **Create.**
6. Click **Close** to close the Data Objects window.
7. Click **Save.**

Creating a Data Object to Store Start Event Arguments

In the next task you will define the start arguments for the Start event. In this procedure, you create the structure to store the arguments.

This procedure involves:

- Creating a business type using the business object option.
- Creating a data object and associating it with the business object.

To create the data object for start event arguments:

1. Open the process definition.

2. Select Business Types from the left panel.

   Note: Notice the other business types created automatically when importing the downloaded JSON, such as ResponseTransactionData.

3. Click **Edit** in the header, which causes the **Create** button to appear.
4. Click **Create,** and select *New Business Object.*
5. On the Create Business Object dialog box, enter the business object name, such as *InitTransaction,* and select the Parent Module as *BusinessData.*
6. Click **Next.**
7. Use the **Add Attribute** button to add these string data types to the business object.

Attribute Name	Data Type
transactionKey	*String*
transactionType	*String*
externalBaseURL	*String*
resourceName	*String*
transactionOwner	*String*
transactionId	*String*
classification	*String*
subclassification	*String*

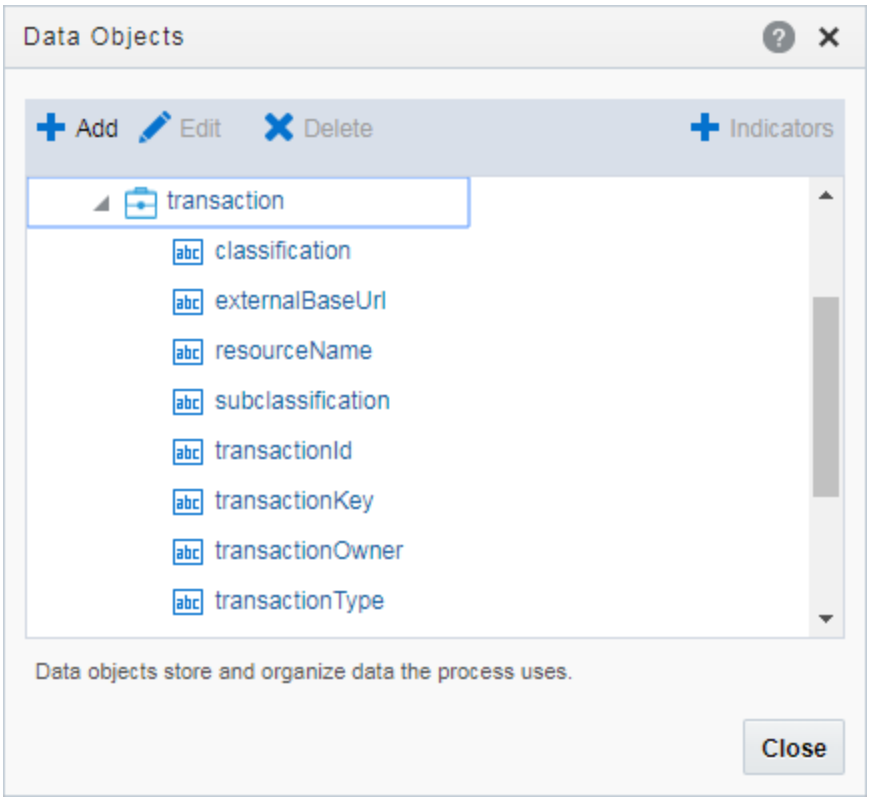
8. Click **Finish.**
9. Return to your process definition diagram.
10. Click the **Data Objects** button.
11. Click **Add.**
12. On the Create Process Data Object dialog box, make these changes:

Field	Value
Name	*transaction*
Data Type	*Business*
drop-down list	*BusinessData.InitTransaction*

Field	Value

13. Click **Create.**

This example illustrates the expanded transaction business data object, with the required string data within it.



Defining Arguments for the Start Event

When a transaction, such as a permit intake application, is submitted, the software instantiates that transaction's workflow process by passing parameters, such as the transaction ID, to OCI Process Automation. The Start event in your process diagram must have arguments defined for these parameters.

To set up the arguments for the start event:

1. Select the Start event in the process definition.
2. Click the Start event, and select *Open Properties.*

   The default view is the General section of the Implementation Properties.
3. In the **How do you want to implement it?** section, select *Define Interface* as the **Type.**
4. Click the pencil icon next to the **Type** field to open the Configure window.
5. Enter an operation name, such as *start.*

6. Add the following rows to the **Arguments Definition.**

    a. Use the **Add** button to add these strings using the values in this table, where each row represents a separate argument you need to create:

Name	Type
TransactionKey	*string*
TransactionType	*string*
ExternalBaseURL	*string*
ResourceName	*string*
TransactionOwner	*string*
TransactionId	*string*
Classification	*string*
Subclassification	*string*

> **Note:** Arguments added to the Start event must be named *exactly* as documented.

> **Note:** You need to complete the output data association if you want the argument values stored in a Data Object.

    b. Click **OK.**

7. Close the properties panel and click **Save.**

Defining Data Associations for the Start Event

The data associations for the Start event capture identifying information about the transaction for initiating the process instance.

In this task, you map the arguments for the Start event to the data object values you entered for the *transaction* object.

To set up the data associations:

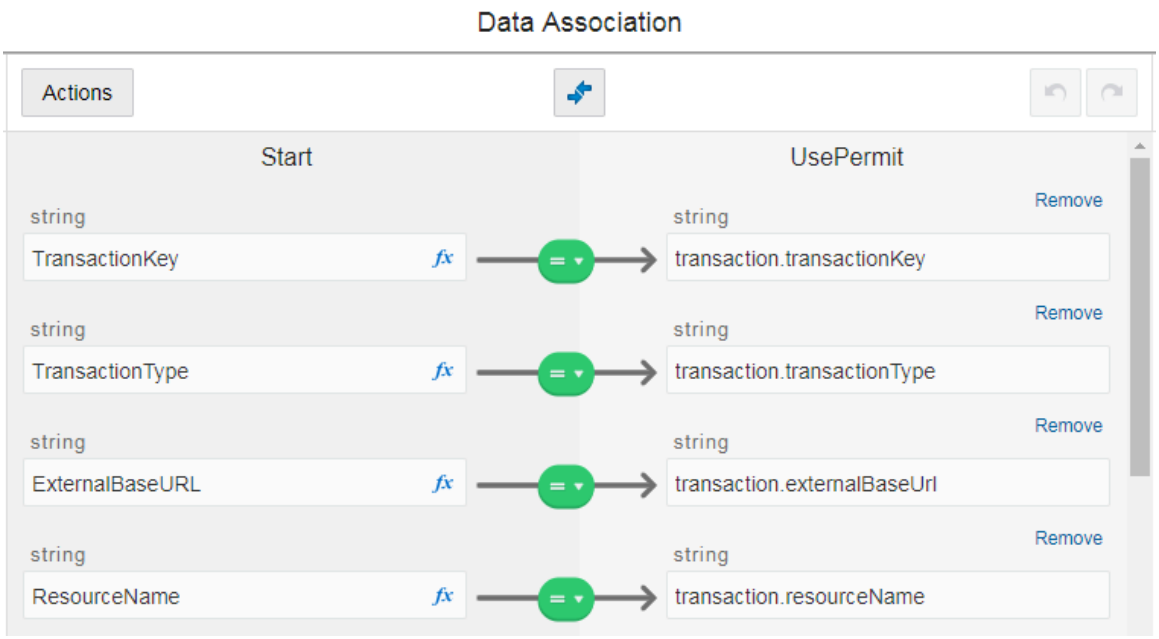
1. Open the process definition and select the Start event.
2. Click the **Data Association** button.

3. Set up the following data associations, mapping the Start event arguments to the appropriate attributes in transaction business object.

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
TransactionKey	*transaction.transactionKey*	The transaction ID.
TransactionType	*transaction.transactionType*	The transaction type ID.
ExternalBaseURL	*transaction.externalBaseURL*	The URL for the Oracle Permitting and Licensing system.
ResourceName	*transaction.resourceName*	The name of the REST API resource.
TransactionOwner	*transaction.transactionOwner*	The owner of the transaction.
TransactionId	*transaction.transactionId*	The transaction ID.
Classification	*transaction.classification*	The classification of the transaction.
Subclassification	*transaction.subClassification*	The subclassification of the transaction.

Source Data (Left side of the map)	Target Data (Right side of the map)	Description

This example illustrates mapping the Start event arguments to the associated data attributes in the transaction business object.



4. Click **Apply.**
5. Click **Save.**

Defining Data Associations for Sending Notifications

The data associations for a notification task define the information that the task sends to the public sector communications center.

> **Note:** Create your email templates in the communications center before you set up integration for notification workflow tasks.

For more information about the communications center, see *Setting Up Communication Templates*.

To set up the data associations:

1. Access the process definition and select the system task.
2. Click the **Data Association** button.

3. Set up the following input data associations:

Source Data (Left side of the map)	**Target Data (Right side of the map)**	**Description**
[event name] For example, *"LNP_WORKFLOW_001"*	*body.eventCode*	The event as defined in the Communications Framework. **Note:** If you have configured the communications connector to call Business Rules Framework to send communications depending on a business rule, the event code refers to the Business Rules Framework event code for the event containing your business rule. The source data string must be in quotation marks, and it must exactly match the identifier of an event. Oracle delivers these communication events, *<Offering>_WORKFLOW_001* through *<Offering>_WORKFLOW_005.* Where "Offering" refers to your offering code, such as LNP for Permits or PNZ for Planning and Zoning.
[template name] For example, *"Application_Accepted"*	*body.templateCode*	The identifier for the template to be used for the email. The source data string must be in quotation marks, and it must exactly match the name of a template in the transaction application.
"LnpRecordKey"	*body.recordFirstKeyName*	The name of the key field.
transaction.transactionKey	*body.recordFirstKeyValue*	The transaction ID.
true or *false*	*body.email*	This Boolean field indicates whether the notification is sent as an email. Enter *true* only if the template is an email template.
true or *false*	*body.notification*	This Boolean field indicates whether the notification is sent as an in-product notification.

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
		. Enter *true* only if the template is an in-product notification template.

> **CAUTION:** Templates are associated with either email or in-system notifications. Be sure to set up the body.email and body.notification values properly. Exactly one of the values must be true. If you want to send both types of notifications, you need to create two notification tasks.

4. Click **Apply.**
5. Click **Save.**

Defining Data Associations for Sending Status Updates

The data associations for a status update task define the information that the task sends to the Oracle Permitting and Licensing system.

To set up the data associations:

1. Access the process definition and select the system task that updates the transaction status.
2. Click the **Data Association** button.
3. Set up the following input data associations:

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
transaction.resourceName	*resourceName*	The unique system identifier for the transaction type.
transaction.transactionKey	*transactionKey*	The transaction ID
[new transaction status] For example: *"Accepted"*	*body.status*	The status to be assigned to the transaction. The source data string must be in quotation marks, and it must exactly match one of the valid statuses for the transaction application.

4. Click **Apply.**
5. Click **Save.**

Defining Data Associations for Retrieving Transaction Base Data

The data associations for a task that retrieves transaction base data provides a structure for storing the retrieved data. In this task you create both input and output data associations.

To set up the data associations:

1. Access the process definition, and select the system task that retrieves transaction base data.
2. Click the **Data Association** button.
3. Set up the following input data associations:

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
transaction.resourceName	*Resource*	The unique system identifier for the transaction type.
transaction.transactionKey	*TransactionKey*	The transaction ID

4. Click **Output.**
5. Set up the following output data associations:

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
body	*BaseData*	This business object contains all of thebase data. Mapping individual fields would be much more complex and is not necessary.

6. Click **Apply.**
7. Click **Save.**

Defining Data Associations for Retrieving Transaction Field Data

The data associations for a task that retrieves transaction field data provides a structure for storing the retrieved data. In this task you create both input and output data associations.

To set up the data associations:

1. Access the process definition, and select the system task that retrieves transaction field data.
2. Click the **Data Association** button.
3. Set up the following input data associations:

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
transaction.resourceName	*transactionResource*	The unique system identifier for the transaction type.
transaction.transactionKey	*TransactionKey*	The transaction ID.

Source Data (Left side of the map)	Target Data (Right side of the map)	Description

4. Click **Output.**
5. Set up the following output data associations:

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
body	*transactionFieldsData*	This business object contains all of the transaction fields. This includes all fields that can be included on the application intake form, whether or not the field exists for a specific transaction. Individual fields are nested within the *items* object. You can't expand the *items* object on this page, but they are available in the expression editor that you use when creating business logic based on transaction data. Mapping individual fields would be much more complex and is not necessary.

6. Click **Apply.**
7. Click **Save.**

Defining Data Associations for Retrieving Transaction Type Data

The data associations for a task that retrieves transaction type data provides a structure for storing the retrieved data. In this task you create both input and output data associations.

To set up the data associations:

1. Access the process definition, and select the system task that retrieves transaction type data.
2. Click the **Data Association** button.
3. Set up the following input data associations:

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
transaction.resourceName	*transactionResource*	The unique system identifier for the transaction type.

4. Click **Output.**
5. Set up the following output data associations:

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
body	*transactionTypeData*	This business object contains all transaction type fields.

6. Click **Apply.**
7. Click **Save.**

Defining Statuses (Outcomes) for Human Tasks

The **Action** property for a human task lists the possible outcomes of the task. The actions you define appear as values in the **Task Status** drop-down list box on the Workflow page where agency staff manages workflow tasks. When the task status is updated on the Workflow page, OCI Process Automation recognizes it as the task outcome and continues to the next step or gateway.

To define status values representing the outcomes of human tasks:

1. Access the process definition and select the human task.
2. Open the task properties.
3. In the **Action** field, enter a comma-delimited list of status values.

   Do not put a space before or after the comma. For example, if the status are *Accept, Reject,* and *More Information,* enter *Accept,Reject,More Information* in the **Action** field.
4. Close the properties window and save.

Defining Conditional Logic for Gateways

In a process map, gateways represent decision points where there is a branch in the process flow. The logic for taking different paths after the gateway is associated with the arrows to the possible subsequent tasks.

An arrow that represents a default branch does not require any logic.

For all other arrows, you need to set up the conditions under which the branch is used. To do this:

1. Access the process definition and select the arrow.
2. Click the pencil icon for the arrow to open the arrow properties.
3. Select the **Conditional Flow** check box.

   This check box is selected for all arrows other than the default arrow after a gateway.
4. Click the pencil icon for the **Condition** field.
5. Use the Expression Editor window to specify the conditions for using this branch.

   The Data Objects tab provides access to the data elements that you can evaluate. Transaction field data (the data from the intake form) is nested within the *items* element under *TransactionFieldsData.*

   In expressions that look for an exact match, take extra care with the spelling, capitalization, and punctuation of values that the expression evaluates.
6. Select the gateway and open the gateway properties.
7. Use the **Order** property to specify the order in which the previous task's outcomes are evaluated for purposes of determining which arrow to follow.

Configuring a Planning and Zoning Human Task to Reference the Assigned Planner Value

If you want to assign the user to a human task, you need to have first setup the Transaction Connector to use the getTransactionAssignee operation.

For more information on setting up the Transaction Connector, see *Setting Up the Transactions Connector*.

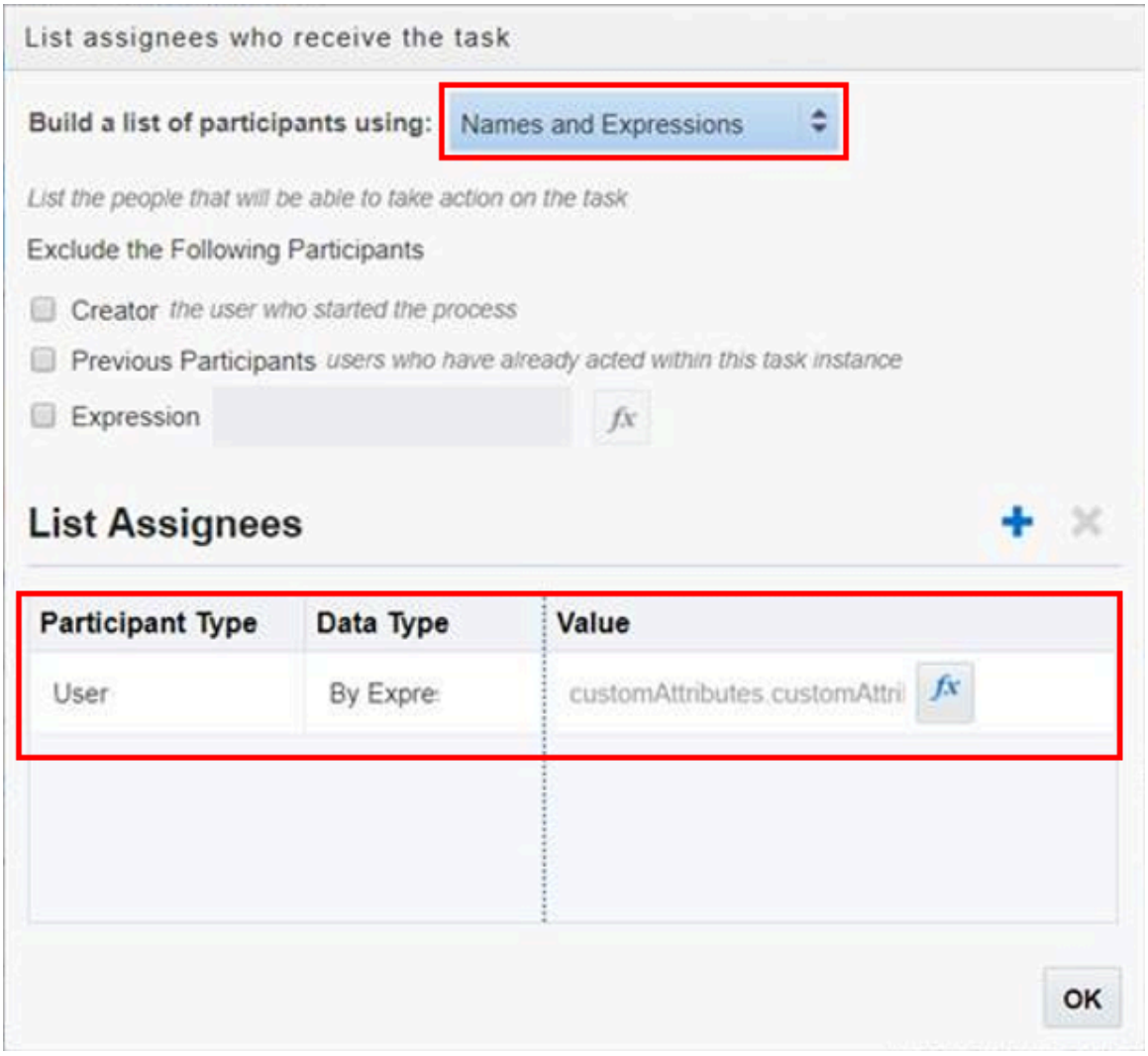
To associate the transaction assignee data to your workflow process:

1. When configuring the workflow process data object, create a data object to store the output of the service call to getTransactionAssignee.
2. Locate the first human task in the process after the service call (Start event), and open the Data Association interface.
3. On the Input Data Association, map the assignee data object to execData.customAttributes.customAttributeString1.

4. For subsequent human task to which you want to reference the assigned planner, set the Assignee(s) value in the General Properties to *customAttributes.customAttributeString1.*

    a. Open the properties for the human task.
    b. Select Implementation > General.
    c. Click the Edit icon for the Assignee(s) field.
    d. Select *Names and Expressions* in the **Build a list of participants using** list.
    e. Click Add for the List Assignees edit box.,
    f. Select *Add User.*
    g. Click in the Data Type column, and select *By Expression*.
    h. Click the **fx** button in the Value column, and select *customAttributes.customAttributeString1.*
    i. Click **OK.**

This example illustrates the settings on the List assignees who receive the task window.



5. Click **Save.**

Implementing the Expiration Feature

If you are implementing the expiration feature for Permits or Planning and Zoning, you need to add these system tasks to the workflow process definition in Oracle Integration Cloud:

System Task	Description
Accepted	Updates the transaction status to *Accepted.* This task should be placed early in the process flow.
Completed	Updates the transaction status to *Completed* or *Certified.* This task should be placed at the end of the process flow.

Using Custom Properties

When you set up workflow, a variety of custom properties are available for implementing various features. This topic describes how to use the custom properties.

Making Custom Properties Available in a Process Definition

The sample definitions provided in the solution package come with custom properties already added. If you are adding custom properties to a custom definition, follow these steps.

To add a custom property to a process definition:

1. Access the Custom Properties dialog box using either of these ways.
    - Open a process definition and click the Custom Properties icon on the right toolbar.
    - Open a process definition, select a task, click the Custom Properties icon on the right toolbar, and click **Add custom property.**
2. Enter the property information in the **Property Name** and **Description** fields in the Custom Properties list:

    > **Note:** You must use the *exact* property names as they appear in the following section. You can provide your own descriptions.

3. Click **OK.**
4. Click **Save.**

Custom Properties

These custom properties are available for Oracle Permitting and Licensing process definitions.

Property	Usage
PSC_LIST_ORDER	Use this property to set the order for human tasks when there are multiple possible paths through the process definition. The order does not affect the workflow process, but it allows users to see the possible future workflow tasks in a logical order. Enter an integer value to indicate the sequence. For example: *2* The Workflow page for a transaction includes an option to view a list of all past, present, and not started human tasks for the permit. The list displays past and present tasks in chronological order. However, the chronology for tasks that haven't been started is not necessarily fixed. The branching logic in a process means that some tasks might be omitted or might occur in a different order depending on permit data or on the outcome of previous tasks. To control the order in which not started human tasks appear, use the PSC_LIST_ORDER custom property. On the Workflow page's list view, tasks that have not yet started are listed in the order you specify. If multiple not started tasks have the same number, they appear in the list in random order.
PSC_FINAL_ACTIVITY	Use this property to identify a human task that isn't allowed to progress when the transaction has a condition that applies the *Prevent Issue or Final* condition rule. Typically, this property is used to identify the final human task in the process definition. You can also use this property on the human task for issuing an application to prevent the application issuance. *Yes* identifies the final activity. A blank value or a *No* value means the *Prevent Issue Final* or condition rule won't be checked. See *Setting Up Conditions* and *Applying Conditions to Applications*. For information about using this property for required documents, see the section, "Custom Properties for Required Documents," in this topic.
PSC_ACTIVITY_TYPE	Use this property to identify the final inspection task in the process. Setting this property is necessary to support the transaction business logic that auto-advances the inspection task when the last inspection is closed. *Inspection* is the only value with related business logic. Leave this property blank for other types of activities. Permit processing includes logic to automatically progress past the final inspection step in the process definition when permit inspections are complete. To enable this functionality, you must identify the final inspection task in the process definition using the PSC_ACTIVITY_TYPE custom property. Further, you must identify the workflow action to apply to that task using the PSC_AUTO_UPDATE_ACTION custom property. This property is also used for plan review completion, planning activities, and assigning or reassigning planners. For information about using this property for required documents, see the section, "Custom Properties for Required Documents," in this topic. The value entered for the Assign Planner task should match what's entered on the Update Workflow action in the Business Rules Framework rule.
PSC_AUTO_UPDATE_ACTION	Use this property to identify the action to take when auto-advancing a task in a process.

Property	Usage
	Setting this property is necessary to support the transaction business logic that auto-advances the inspection task when the last inspection is closed. It's also used to auto-advance an assigned planner. Use the exact action name as specified in the Action property for the human task. Take extra care with the spelling, capitalization, and punctuation of the action name. For example: *Approve* or *Assign Planner*. **Note:** Security roles related to the Mobile Inspector app and the Plan Reviewer should be included in the task swimlane mapping.
PSC_UNRESTRICTED_ACTIONS	Identifies actions that can be taken for the task even if there is logic preventing the task from advancing. Use this property when the task includes possible outcomes that return to a previous task rather than advancing. For example, if the final inspection can be advanced with an action of “Approve” or returned with an action of “Needs More Information,” then use the PSC_UNRESTRICTED_ACTIONS property to make “Needs More Information” an unrestricted action. This allows users to take the “Needs More Information” action, even though the logic that prevents the final inspection task from advancing disallows other actions. When there are multiple unrestricted actions, separate the actions with commas but no spaces. This is the same format used in the Actions property where you define all of the available actions for a human task. For example: *Needs more Info, Proceed, OK*
PSC_MIGRATION_TASK	Use this property to map the transaction statuses of your migrated legacy data to the correct Oracle Permitting and Licensing workflow user tasks. You must set this property for your process definition workflow user tasks before running the Prepare Migrated Data for PSCR process. See *Generating Workflow Status for Migrated Transactions*. The status of the migrated legacy transaction. For example: *SUB*
PSC_REQUEST_ACCEPT_INFO_HIDE	Use this property on human tasks to hide the Request Details button on the two-panel workflow tab in the transaction details pages. Setting the value to *True* or *TRUE* hides both the Request Details button and the Accept Info button. Example: *True* For more information on the Request Details button and Accept Info button see *Using Workflow*.
PSC_REQUEST_INFO_LABEL	Use this property on human tasks to provide a custom label for the **Request Info** button that appears by default on the two-panel workflow tab in the transaction details pages. Example: *Request Details*

Property	Usage
	Note: You can request more information at different stages of the workflow by configuring the **Request Info** drop-down list with various options. When the drop-down list is configured, you cannot customize the **Request Info** label. You will also have an option of sending an e-mail to interested parties for more information. To enable the drop-down list, you need to provide the menu items for the list, and whether the email option will be enabled. Use the following syntax: `(<menu item>, ENABLE_COMM/DISABLE_COM)` Where ENABLE_COMM enables the email communication to be invoked, and DISABLE_COM does not display the email option. For example: (Pending Deposit, ENABLE_COMM),(Missing Details, DISABLE_COMM),(Revision Required, DISABLE_COMM)
PSC_ACCEPT_INFO_LABEL	Use this property on human tasks to provide a custom label for the **Accept Info** button that appears by default on the two-panel workflow tab in the transaction details pages. Example: *Accept Details* **Note:** After you configure the **Request Info** and **Accept Info**ed only when the information is accepted using the **Accept Info** option. When the drop-list is enabled, you cannot customize the **Accept Info** label. You will also have an option of sending an e-mail to interested parties for more information. To enable the drop-down list, you need to provide the menu items for the list, and whether the email option will be enabled. Use the following syntax: `(<menu item>, ENABLE_COMM/DISABLE_COM)` For example: (Deposit Received, ENABLE_COMM), (Details Received, DISABLE_COMM) , (Revision Complete, DISABLE_COMM)
PSC_WFACTION_COMMENT	Set this for any human task in the process definition. This property enables you to control whether comments are required when updating the **Status** drop-down list for a human task on the Workflow tab of the transaction details. **Note:** The **Status** drop-down list is populated by the **Action** field in the General properties of a human task. Options are: • *MANDATORY:* Any status update requires a comment to be entered. • *OPTIONAL:* A comment can be added if desired but not required for any status update.

Property	Usage
	• *SELECTIVE:* A comment is required for one or more particular status values. For example, if you only want to require a comment when the task status has been set to *Reject* to explain why the application did not meet requirements. You specify the status values requiring a comment using the PSC_WFACTION_SELECTIVE property.
PSC_WFACTION_SELECTIVE	Set this for any workflow human task in the process definition where the PSC_WFACTION_COMMENT property has been set to *SELECTIVE*. Add status value(s) for the selected workflow task where you want the agency user to provide a comment. When the specified task status value is selected, a comment becomes required. If there are multiple statuses for which you want to require a comment, enter a comma-separated list. For example: *Reject,Requires More Information*
PSC_BRF_EVENT_CODE	Specify the event code for the Business Rules Framework event to call for validating the transaction information to ensure that all requirements are met for the workflow process to advance to the next task or to trigger other business rule actions for that event. For example, include the event code for the event where the business rule triggering a Stop Process action determines if a permit or business license should be issued. The event codes for the permit and business license issuance events are: • *Before Permit Issuance* (LNP-BEFORE-ISSUANCE) • *Before Business License Issuance* (BL-BEFORE-ISSUANCE)
PSC_BRF_ALL_WF_ACTIONS	When you specify PSC_BRF_EVENT_CODE, PSC_BRF_ALL_WF_ACTIONS determines whether to invoke the Business Rule Framework event in all situations or selected. Enter values *Yes* or *No*. • *Yes:* to invoke for all workflow actions (the default). • *No:* to invoke only if the workflow status isn't listed in the PSC_UNRISTRICTED_ACTIONS custom property.
PSC_ALTERWF_OPT	Enables a workflow task to be available to be selected during the alter workflow process. For more information on this option see, *Enabling Tasks for Altering Workflow and Reopening Applications*
PSC_REOPENWF_OPT	Enables a workflow task to be available to be selected during the reopen application process. For more information on this option see, *Enabling Tasks for Altering Workflow and Reopening Applications*

Custom Properties for Required Documents

Agencies can set up business license, permit, and planning applications to require documents at different steps throughout the application process. To enable validations of required documents at the Application Acceptance, Application Issuance, and Application Completion workflow tasks, use the custom properties as described here. For more information about settings to enforce document requirements, see *Setting Up Required Documents*.

Example Workflow Task	PSC_ACTIVITY_TYPE	PSC_FINAL_ACTIVITY
Application Acceptance	*Accept Application*	*Yes* or *No*
Application Issuance	*Issue*	*Yes* or *No*
Application Completion	Enter any value.	*Yes*

Mapping Workflow Swimlanes to Roles

This topic describes how to assign security roles to the swimlanes in your workflow process definition.

In workflow process definitions, swimlanes represent roles. The swimlanes are the horizontal rows in your diagram indicating which user is responser for various tasks. After the OCI Process Automation application containing the process definition is activated, use the Roles functionality in OCI Process Automation to map the swimlanes to roles. The mapping applies to all process definitions within the OCI Process Automation application. In a typical configuration, each swimlane is mapped to a role.

A swimlane is typically associated with security roles, and it can be associated with multiple roles if needed. It can also be associated with one or more individual users if that approach is more applicable. A swimlane determines who is responsible for carrying out a task.

For example, if a swimlane is for plan review, you would add the PSC Plan Reviewer role to that swimlane. Likewise, if another swimlane is for inspecting, you would add all the roles that apply to that swimlane, such as PSC Building Inspector and any roles related to the Mobile Inspector app.

Note: When supervisors assign or reassign tasks, they can only assign the task to agency staff associated with security roles that are assigned to the swimlane in the underlying workflow process definition. The swimlane that contains the Start event node needs to be mapped to the PSCR Submitter Group. The procedure below uses that scenario as an example to illustrate the process used to map a security role to a swimlane.

Note: If a task is calling the Business Rules Framework and a business rule includes an action to update workflow, then the swimlane the task resides in *must* be assigned the *PSCR Submitter Group* or an individual user associated with this group.

To map swimlanes to roles:

1. Access the Workspace area of OCI Process Automation by selecting *Workspace* from the Navigator menu.
2. In the Workspace, select *Roles* from the Navigator menu.
3. At the top of the work area, select Application, to display the roles (swimlanes) for the applications.
4. In the Scope column, locate your process application and view all the roles included in that application.

   The *Process Owner* and *Process Reviewer* roles are part of all applications. They aren't specific to Oracle Permitting and Licensing functionality and you don't need to map roles to them. Other roles in the list for your application are the swimlanes in your process definitions. You will need to map the swimlane to an OPAL (PSCR) role (or *groups* as they are referred to in OCI Process Automation).

5. Add the delivered role *PSCR Submitter Group* to the swimlane that contains the Start event for your process model:
    a. Select the swimlane that contains the Start task in your process definitions.

       In the delivered Solution Packages that Oracle provides, this swimlane is labeled *Applicant.* This swimlane applies to the user submitting a transaction, such as a permit application.
    b. In the **Add members to this role** list for the selected swimlane, select Groups in the **Search by** field.
    c. In the dialog box for adding members, search by *Groups* for *PSCR Submitter Group.*

       A group in OCI Process Automation is equivalent to a role in the Oracle Permitting and Licensing system.
    d. In the search results, select *PSCR Submitter Group* and then click to assign the role to the swimlane and return to the list of swimlanes.
    e. Expand the Permissions section, and confirm the Action column is set to *Manage.*
6. Click **Save.**
7. Repeat these steps for all required roles per swimlane.

For example, in addition to the PSCR Submitter Group being added to the swimlane with the Start event (Applicant), the following identify some samples of how other swimlanes might be mapped, depending on the offering and the process definition.

Offering	Swimlane Mapping
Permits	Applicant: • PSCR Submitter Group (group) • PSCR Proxy User for OIC Inspection Supervisor: • PSC Inspections Supervisor (group) • PSC System Administrator (group) Permit Technician: • PSC Permit Technician (group) • PSC System Administrator (group) Plan Review Coordinator: • PSC Plan Reviewer (group) • PSC Planning Coordinator (group) • PSC System Administrator (group)
Planning and Zoning	Applicant: • PSCR Submitter Group (group) • PSCR Proxy User for OIC Planning Assistant: • PSCR Submitter Group (group) • PSC Planning Assistant (group) • PSC System Administrator (group) Principal Planner:

Offering	Swimlane Mapping
	• PSCR Submitter Group (group) • PSC Principal Planner (group) • PSC System Administrator (group)

Preparing the Process Definition for Use

This topic describes how to publish and activate the process definition so that it can be referenced and used for intake forms.

Preparing process definitions for use involves:

- Versions
- Activation
- Linking to a transaction type

Working with Versions

Your first iteration of a process definition is version 1.0. After you activate that process definition, you would then need to create a new version explicitly before rolling out new changes. The new version would be 1.0, 2.0, or whatever your numbering scheme is at your agency. You can have numerous versions of an application in design mode if needed. You create a new version by clicking the version number for that process application to display a drop-down menu and clicking **New version.**

> **Note:** The version you reference for your transaction type (such as in the Workflow Setup section on the Permit Type page) applies to all new submitted transactions, while the previous version will continue to apply to the transactions that were submitted while that version was referenced by the transaction type.

For more information on OCI Process Automation versions, see *Work with Versions.*

When working with versions, keep these items in mind:

- Do not reuse the same version number used on a previous version of the process definition. If you reuse the same version number when you activate the application, *all open process instances* using that version will be terminated and sent to a status of *complete* regardless of where they actually are in the workflow. To prevent this, make sure to use a new version number for each newly activated version.
- When you activate and associate a new version number to a transaction type definition, such as a permit, only the transactions submitted *after* the new version of the workflow process definition has been applied can take advantage of the changes made in the newest version of the process definition. All in-process transactions continue to use the workflow process definition version number in place when those applications were submitted. For example, if you make a change to the role or user ID assigned to a swimlane in version 2 of a process definition, only the intake forms submitted *after* version 2 has been associated with the transaction type can take advantage of the swimlane assignment change. All intake forms currently being processed continue to use swimlane assignments defined in version 1 of that workflow process definition.

- Do not deactivate a previous version of a process definition because in-process transactions may still be using it.

Activating Process Definitions

Activating an application makes any new or modified process definitions within that application available to associate with an intake form for a transaction type definition. When you've completed your changes to a version, click **Activate.**

For more information see *Activate Applications.*

Linking a Process to a Transaction Type

When an application has been published and activated, you can link it to a transaction type. On the transaction type page, such as the Permit Type page, use the Workflow Setup section to specify the workflow process definition to use for that transaction type. There you specify the:

- Application
- Version
- Process Definition

For more information, see:

- *Setting Up Permit Types*
- *Setting Up Planning Application Types*
- *Setting Up Business License Types*

Monitoring Workflow Transactions

This topic describes how to use the Manage Process Instances page to resolve discrepancies between the Oracle Permitting and Licensing and OCI Process Automation.

Working with Workflow Transaction Logs

During normal transaction processing, it's possible that either OCI Process Automation or Oracle Permitting and Licensing can become momentarily unresponsive. While a rare occurrence, you need to be able to resolve any transactions that occurred during the down time to make sure the workflow engine instance and your Oracle Permitting and Licensing offering are synchronized.

For example, if OCI Process Automation is unresponsive for a five minute period according to your system logs and notification system, you can use the Manage Process Instances page to isolate any transactions that occurred specifically during that time to make sure individual transaction statuses are in sync with the associated workflow process.

> **Note:** In some cases, if OCI Process Automation is momentarily unavailable, a process instance won't be assigned to an application. In this situation, a background process periodically checks for missing process instances and assigns them accordingly. While the application has no process instance assigned, its status will be set to *Pending Submittal.* For more information on the Pending Submittal status see "Managing Application Activity" in *Managing Applications* and *Managing License Applications*.

Using the Manage Process Instances Page

Use the Manage Process Instances page to search for transactions created during a specific time range.

Access the Mange Process Instances page by selecting **Workflow and Transaction Log** > **Manage Process Instances**.

Page Element	Description
Which system is restored	Select the system that was not available for a period of time and needed to be restored from a back up. Options are: • *PSCR:* Select if the Oracle Permitting and Licensing system was down. • *OIC:* Select if the Oracle Integration Cloud instance was down. If one of the systems is unavailable for a given period and needed to be restored, then the other system becomes the most current source of transaction status for any manual synchronization.
From/To	Use the date/time controls to specify a time range for isolating the affected transactions. Click **OK** to run your search.
Select Multiple	Enables you to select multiple rows of your returned search values. **Note:** Currently, only active Code Enforcement processes enable the section of multiple items. Only those rows display a check box to select.
Mark Inconsistent	View by pressing the **Select Multiple** button. This applies only to active Code Enforcement transactions. Press to indicate the selected active transactions are inconsistent due to system downtime or outage. For the selected processes, a message appears on the Workflow tab letting the agency user know that a discrepancy needs to be resolved. A **Details** button will also appear on the Workflow tab for privileged users to click to display the Manage Process Instances page, which can be used to resolve the issue, such as by resubmitting, altering workflow, and so on.
Mark Resolved	View by pressing the **Select Multiple** button. This applies only to active Code Enforcement transactions. After a process has been marked as inconsistent, and later resolved, click **Mark Resolved** to indicate the process issue is taken care of. Doing so removes the inconsistent message on the Workflow page and the **Details** button.
Workflow Instance ID	The workflow instance ID assigned for a specific transaction by the OCI Process Automation system when that transaction is submitted.
Process Status	The current status of the process, such as *CANCELLED, OPEN, COMPLETED,* and so on.
Transaction	The unique ID of a specific submitted transaction, which is comprised of the topic type and the auto number rule.

Page Element	Description
Transaction Status	The current status of the individual transaction according to the corresponding workflow process definition, such as *Submitted, Inspection, Plan Review,* and so on.
Process Definition	The workflow process definition in OCI Process Automation to which the specific transaction is associated. This value is comprised of the instance name and the workflow process application name.
Transaction Update Date	The last time the transaction was updated.
Process Update Date	The last time the process was updated.

Using the Details Page

Use the Details page to drill into a specific transaction and take action to resolve any discrepancies between the OCI Process Automation and Oracle Permitting and Licensing systems. The Details page contains the transaction history in sequential order between the two systems for a given transaction.

Access the Details page by clicking on any row in the grid on the Manage Process Instances page.

At the top of the Details page the Oracle Integration Cloud and the Oracle Permitting and Licensing sections display a high-level view of the current status for that transaction in the respective systems.

Depending on the information displayed, you may elect to perform different actions, which appear as buttons on the right hand side of the Details page.

Page Element	
Done	Closes the Details page with no action taken. Equivalent to Cancel.
Cancel Workflow	Cancels the transaction associated with the underlying workflow so the transaction can be resubmitted. All previous work would be lost. This button would appear for a structured process if the Oracle Permitting and Licensing system has been restored from outage and had lost application details associated with a given OIC process instance. As in, the transaction record ID is not known.
Alter Workflow	Enables the system administrator to alter the workflow manually to ensure both systems are in sync in the event of an momentary unavailability of either system. For example, this button may appear when OCI Process Automation has been restored from outage and it has lost the details related to one or more tasks or stages. For the structured processes, such as those used for permits, you can select the desired activity name and set its status. For dynamic processes, such as those used for Code Enforcement, select the event, such as the global task, to reactivate the process stage. For more information on the options when altering workflow, see the section below "Altering Workflow."
Resubmit Application	Enables the system administrator to resubmit an application.

Page Element	
	Appears if OCI Process Automation has been restored from outage and it has lost the process instance belonging to the transaction, as in the OCI Process Automation process ID is not found.

Page Element	Description
Process ID	The workflow instance ID assigned for a specific transaction by the OCI Process Automation system when that transaction is submitted.
Transaction ID	The unique ID of a specific submitted transaction, which is based on the auto number rule.
Applicant ID	The unique ID of the applicant who submitted the transaction intake form.
Transaction Type	The transaction type as defined on the Transaction Type page. For example, for a permit this will be the Permit Type value.
Last Task	The last task in the workflow process definition that the transaction has reached. (Applies only to the OCI Process Automation system.)
Last Action Taken	The action taken on the last task. (Applies only to the OCI Process Automation system.)
Transaction Status	The current status of the transaction reflected in the Oracle Permitting and Licensing system, such as *In Process, Accepted,* and so on.
Inconsistent	Applies to Code Enforcement processes only. Use to mark a process as inconsistent because of discrepancies caused by downtime or system outage. A message appears on the Workflow tab letting the agency user know that a discrepancy needs to be resolved. A **Details** button will also appear on the Workflow tab for privileged users to click to display the Manage Process Instances page, which can be used to resolve the issue, such as by resubmitting, altering workflow, and so on.

Page Element	Description
Service Type	Indicates what service handled the request. Options are: • *Oracle Permitting and Licensing:* Indicates the Public Sector system. • *OIC:* Indicates the Oracle Integration Cloud instance.
Event	Applies to Code Enforcement only. Displays the information for the event that activated the current stage, such as the global task.
Task Information	The task information associated with the task defined in the workflow process definition.

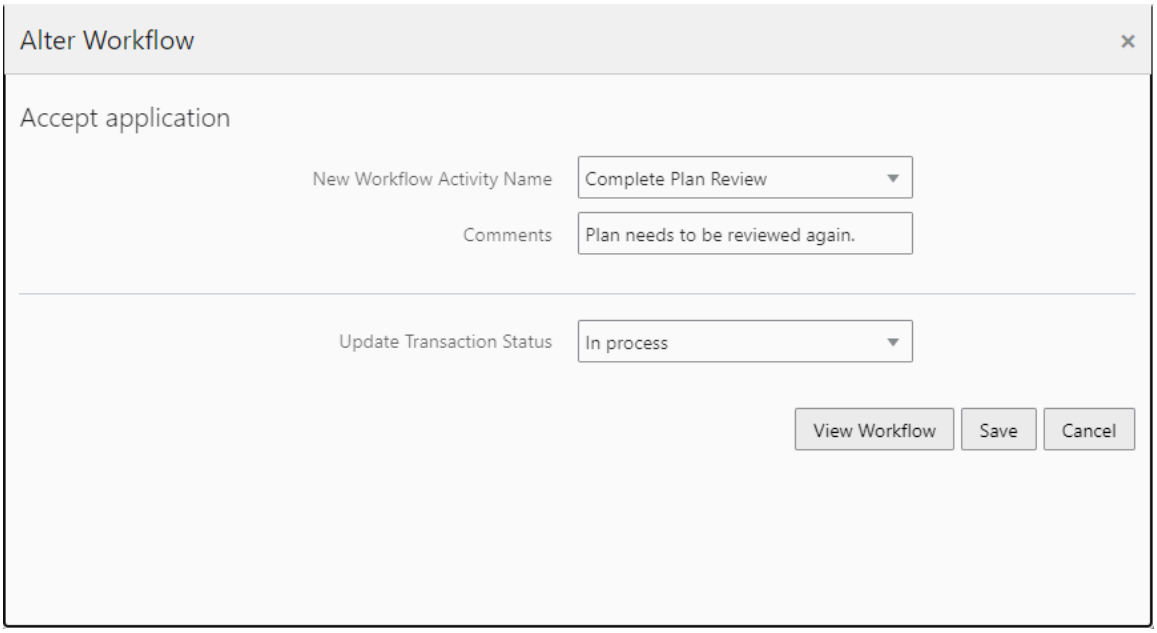
Page Element	Description
Assigned By	The user name of the individual to who assigned the current task to the assignee.
Assigned To	The user name of the individual to whom the current task is assigned.
Updated By	The user name of the individual who updated the task.
Task Creation Date	The date and time the task in the workflow process definition was initiated for the current transaction.
Last Action Taken	The last action taken as reflected in the workflow system.
Transaction Status	The status of the transaction reflected in the Oracle Permitting and Licensing system.
Last Updated Date	The date and time when the transaction was last updated.

Altering Workflow

Click **Alter Workflow** when you need to intervene in the process flow manually. You can change the process flow of an instance that is currently suspended because of a problem, or move an instance that is running to another activity because of a specific reason.

For example, if an activity is failing because of the value of the data objects and instance attributes, then you can modify them and retry running the current activity again. Or, if either OIC or Oracle Permitting and Licensing is down at some time, you may need to analyze and resubmit the process instance or move it along as needed.

When working with a structured process, such as those used in Permits, Planning and Zoning, and Business Licenses, you can use the Alter Workflow dialog box to select a new activity to switch to from the current activity and set the transaction status. To see the workflow process diagram, click **View Workflow.**



When working with a dynamic process, such as those used in Code Enforcement, you can use the Alter Workflow dialog box to select the event that will trigger the desired stage.

Using the Workflow List

Supervisors and system administrators can use the Workflow List page to review in-process workflow instances and manage the process as needed, such as altering workflow.

To access the Workflow List page, select the **Workflow List** tile on the agency springboard.

Managing Worklists

This topic describes how to set up and manage various worklist features.

Displaying Faulted Processes and Alerted Tasks

You can display workflow processes with issues and alerted tasks in the agency user's worklist. Using the Faulted Transactions and Alerted Tasks tabs, the agency users gains quick access to this information, and they can take action as needed. Populating these worklist tabs requires running an ESS job on a regular basis.

To populate the Faulted Transactions and Alerted Tasks tabs:

1. Navigate to Setup and Maintenance.
2. Select the offering, such as *Public Sector Permits.*
3. Go to the System Administration functional area for the offering.
4. Select this task: *Managing Faulted Workflow Processes.*
5. On the Load Public Sector Faulted Workflow Processes Job page, select the Schedule tab and set up a schedule to run the ESS job to suit your business requirements.
6. Click **Submit**.

Enabling the Enhanced Worklist

You can enable the enhanced worklist feature, which flattens the worklist layout and task lists so each data element occupies its own column. The enhanced list layout improves searching capabilities and allows you to use any column for sorting.

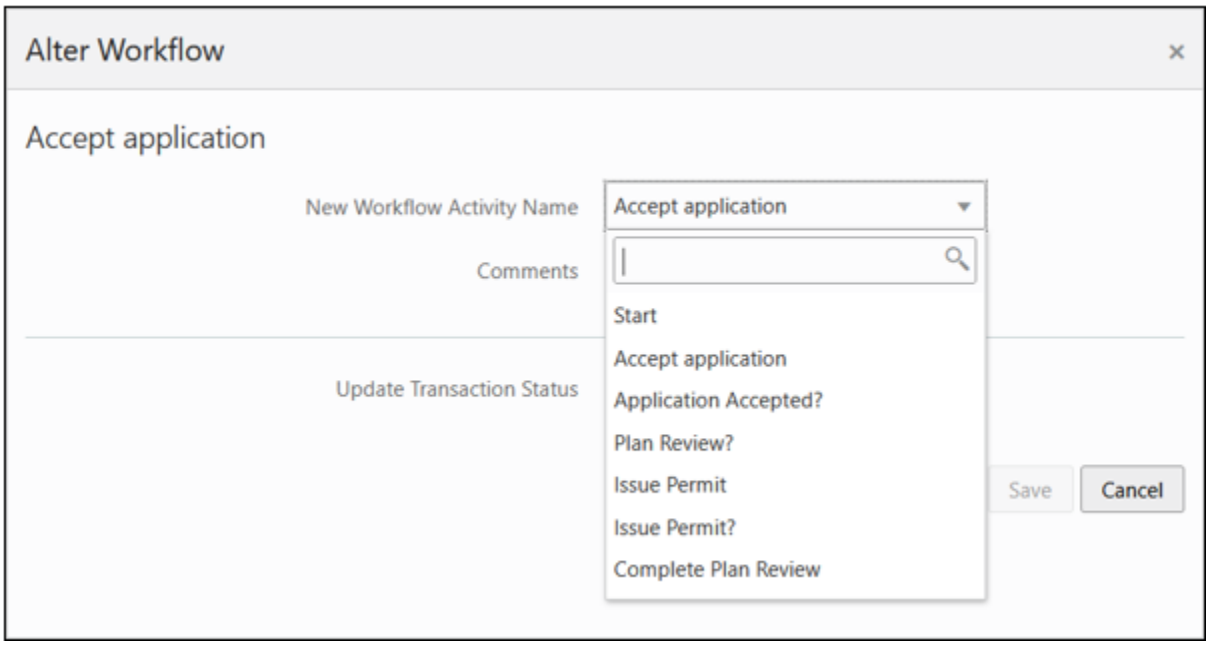
To enable the enhanced worklist:

1. Navigate to Setup and Maintenance.
2. Select the offering, such as *Public Sector Permits.*
3. Click the Change Feature Opt In link.
4. Select the Features icon for System Administration.
5. Select the **Enable** check box for the Enhanced Worklist feature.
6. Go to the Initial Setup functional area for the offering.
7. Select this task: *Run Load Public Sector Workflow Tasks Job.*

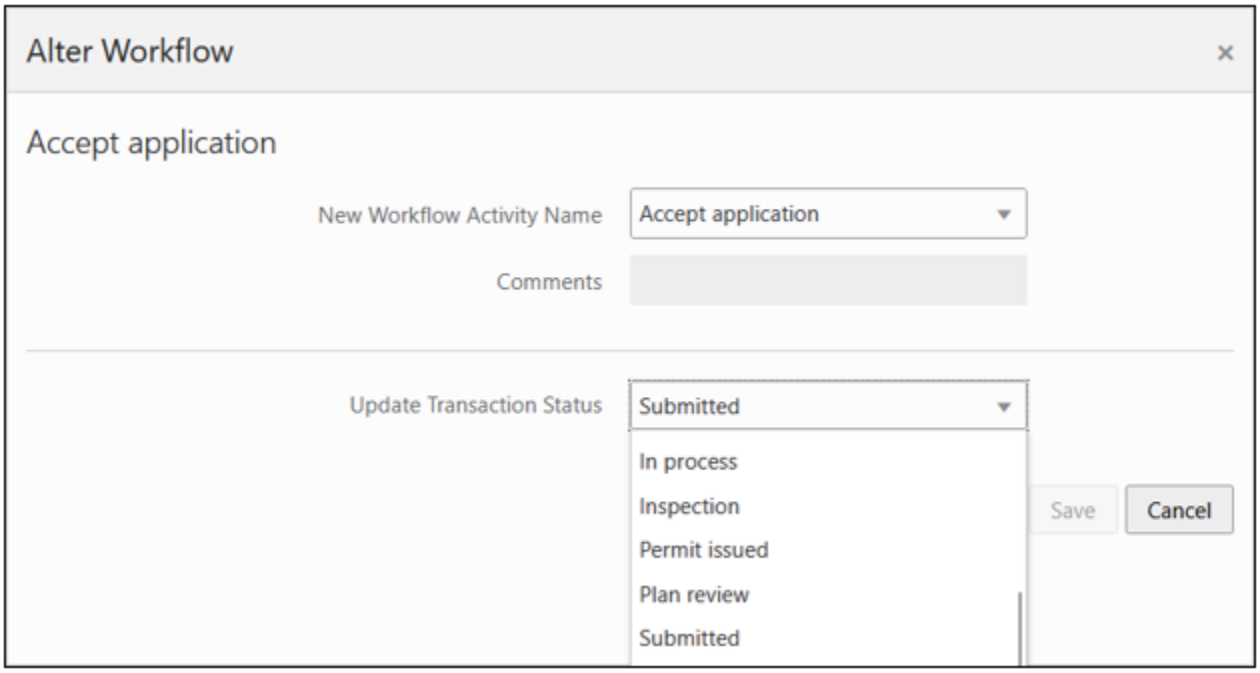
Enabling Tasks for Altering Workflow and Reopening Applications

This topic describes how to enable and configure the ability to display workflow tasks selectively for users when altering workflow or reopening applications.

When altering workflow or reopening applications, agency users can have difficulty selecting and assigning the appropriate workflow task and transaction status. By default, numerous workflow tasks display for the user to select.



Likewise, users can select from all available transaction statuses.



To reduce the possibility of human error while streamlining the user experience, you can:

- Enable tasks to display selectively for the users to select from the **New Workflow Activity Name** drop-down list when altering workflow or reopening an application.
- Hide the **Update Transaction Status** drop-down list from appearing by updating required transaction statues within the workflow process definition.

Setting up this configuration option involves:

1. Enabling the Functional Setup Manager opt-in feature *Selective Enablement of Alter Workflow*.
2. Updating your structured workflow process definitions to incorporate these custom properties:

- PSC_ALTERWF_OPT for selected tasks to be enabled for alter workflow.
- PSC_REOPENWF_OPT for selected tasks to be enabled for reopening applications.

Note: This feature applies only to structured workflow processes. It doesn't apply to dynamic processes used by Code Enforcement.

Enabling Selective Workflow for Alter and Reopen

To enable selective workflow for alter and reopen:

1. Open Functional Setup Manager.

   For example, select *Setup and Maintenance.*
2. Select your offering and click the Change Feature Opt In link in the Functional Areas box.
3. On the Opt In page for your offering, click the Features icon for the top-level offering in the features grid.

   For example, for Permits click the Features icon at the *Public Sector Permits* level.
4. In the feature grid, select **Selective enablement of alter workflow.**

   Note: Selecting this option turns this option on for alter workflow *and* reopen application.
5. Click **Done**, then click **Done** again.

Modifying Process Definitions for Selective Workflow for Alter and Reopen Display

Add these custom properties to workflow tasks to enable a task for alter workflow or reopen application.

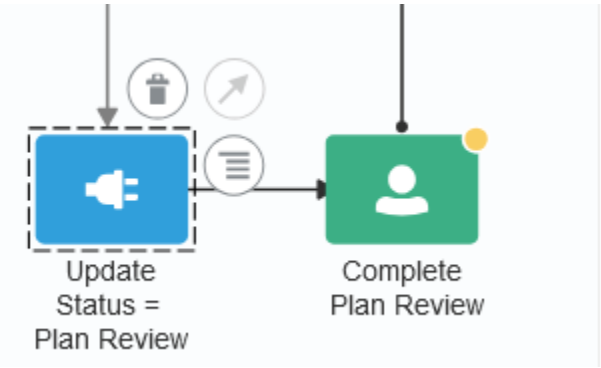
Custom Property	Description
PSC_ALTERWF_OPT	Enables a workflow task to be available to be selected during the alter workflow process.
PSC_REOPENWF_OPT	Enables a workflow task to be available to be selected during the reopen application process.

After you enable your system for this feature, you need to add the PSC_ALTERWF_OPT and PSC_REOPENWF_OPT custom properties to each task you want to display for users to select.

If a human task has a system task before it that sets the transaction, you can add the custom property to that system task, setting the appropriate transaction status, such as *Plan Review*, and the workflow engine will update the status and restart the workflow automatically at the next human task. Use the following syntax:

```
(transaction status)
```

For example:

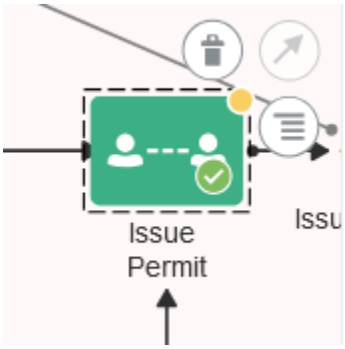


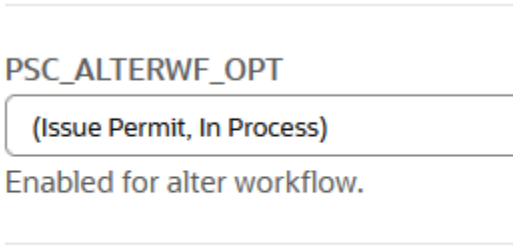
This approach also applies if a human task doesn't require a transaction status update.

If a human task doesn't have a system task before it and a transaction status update is required, you can set both the task name and the transaction status, using the following syntax:

```
(task name, transaction status)
```

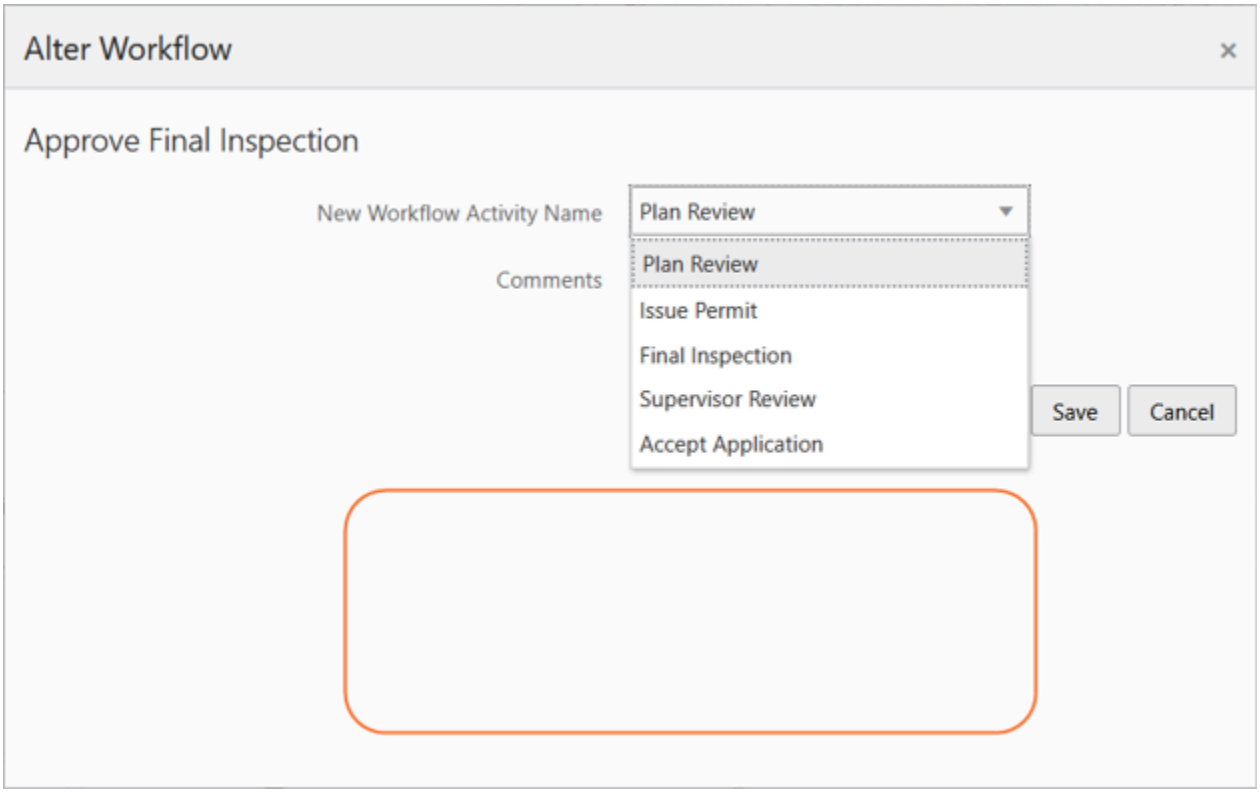
For example:





Only the tasks to which you add values for the PSC_ALTERWF_OPT and PSC_REOPENWF_OPT custom properties are enabled for either alter workflow or reopen application. Any task with a blank PSC_ALTERWF_OPT or PSC_REOPENWF_OPT custom property is ignored by alter workflow and reopen application.

At run time, the selected list displays for the user in the **New Workflow Activity Name** drop-down list, and because the transaction status has been set within the process definition itself, the **Update Transaction Status** drop-down list doesn't appear.



Note: The custom properties PSC_ALTERWF_OPT and PSC_REOPENWF_OPT use the same syntax and behave the same within the respective features.

Working with Parallel or Inclusive Gateways

When working with parallel/inclusive gateways:

- During alter workflow, tasks inside parallel/inclusive gateways won't be visible from outside the gateway. To alter workflow to tasks inside parallel/inclusive gateways, set the alter to the gateway start node.
- Once inside the Parallel/Inclusive gateway, tasks outside the gateway won't be available until the gateway is exited.

4 Setting Up Dynamic Workflow

Code Enforcement Workflow Basics

This topic introduces you to the elements in a OCI Process Automation dynamic process definition used in Oracle Permitting and Licensing Code Enforcement offering.

Code Enforcement Workflow Overview

Adding workflow to Code Enforcement transactions enables you to automate the progression through the stages and activities and incident or case process flow. Oracle Permitting and Licensing offerings use the OCI Process Automation for designing workflow process definitions and running the workflow engine that drives transaction automation. Before you begin implementing workflow for Code Enforcement, it is imperative that you become familiar with the *Processes* feature in OCI Process Automation.

For more information on the OCI Process Automation Processes feature, see *Using Processes in Oracle Integration*.

How Code Enforcement implements workflow is a little different than the way other Oracle Permitting and Licensing offerings do. Permits, Planning and Zoning and Business Licenses use a *structured* process design, which is suitable for more linear, sequential transaction flows. The Code Enforcement transaction flows can contain stages and activities that don't necessarily occur in a set order, with some may occurring at the same time, while others may not occur at all.

Because of the non-sequential nature of a typical Code Enforcement transaction flow, you will create a *dynamic* process design. The dynamic process design is a departure from the structured "step 1, step 2, step 3" approach to process design. With a dynamic process definition, you define the stages and activities of the process flow, but you don't define any particular order. You define conditions that drive *when* or *if* a particular stage or activity becomes activated, providing the flexibility the Code Enforcement offering requires.

For more information on dynamic processes, see *Develop Dynamic Processes*.

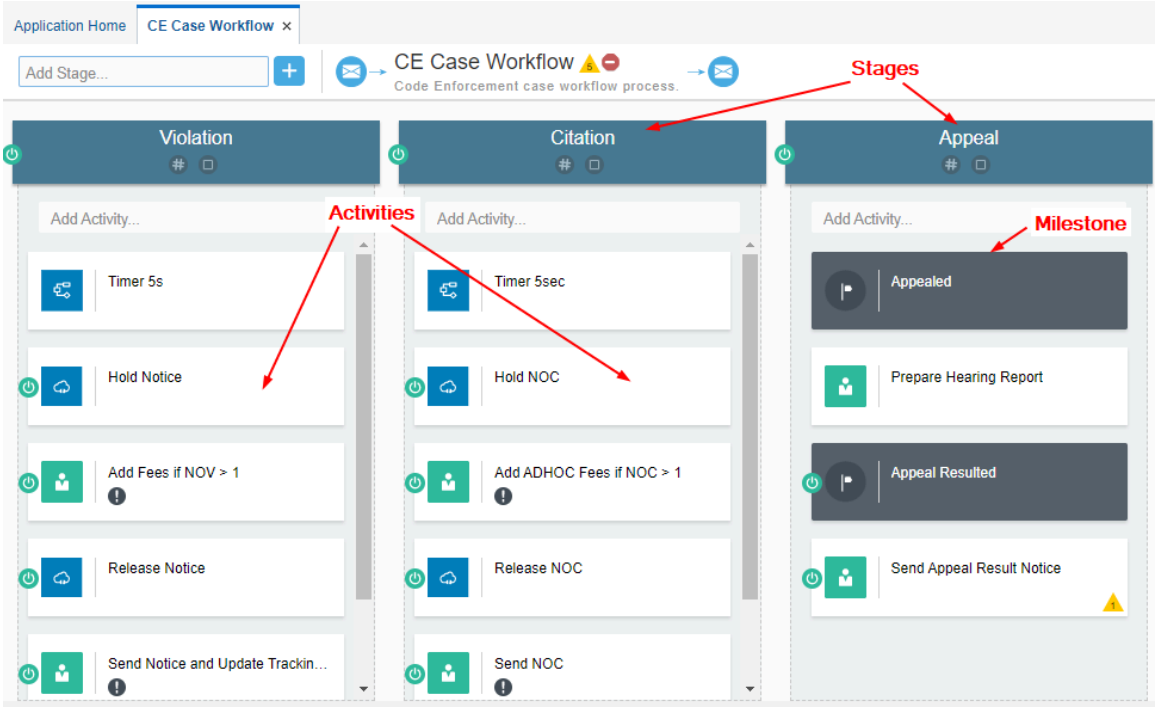
> **Note:** Oracle provides a Solution Package with sample workflow configurations. It is highly recommended that you clone these samples and use them as starting points to create your own workflow.

Working with a Dynamic Process Definition

This topic provides an overview of the main elements in the dynamic process design used for the Code Enforcement offering and describes the top-level configurations you can make.

The sample process definitions provided in the solution package contains examples of all the elements that can be used for a Code Enforcement. The topics in this chapter go into more detail on each element and describe how to configure the element within the process, while pointing out what you should not change. With all of the elements, the power and flexibility they provide can be achieved by configuring the conditions that determine when they become active.

This example illustrates the main elements of a Code Enforcement dynamic workflow process. Details are in the surrounding text.



Process-Level Settings

In the process-level settings, which you can access in the header of the process definition should be used as delivered in the solution package samples.

The input data is mapped to the data stored in the business type. Click the **Input Data** button to display the Start the Dynamic Process window. The process for Code Enforcement should be started with the **With Data Only** option selected. The Interface Arguments reflect the data stored in the business type.

When editing the process-level settings, you can change:

- Name (which you create when cloning the sample)
- Description
- Roles

The Termination condition set at the process level enables the entire process instance to be stopped in the event that the case status is set to closed.

The roles that you include at the process level will be inherited and can be used in all of the elements within that process, such as stages or activities. At the stage and activity level, you can add more specific security settings (roles and users) as needed. You can create roles for the process and map them to existing PSC roles, such as *PSC Code Enforcement Officer* or to specific users if required.

Stages

Stages enable you to organize activities into phases of a process. Stages can run at the same time or one after another. Examples of stages in the Code Enforcement process definition include Violation, Citation, Appeal, and so on. Each stage represents a different phase of the workflow with its own set of activities.

See *Setting Up Stages*.

Activities

Activities represent actions or tasks that need to be completed within the process. Activities can be carried out by a human, or they can be automatically completed by calling another process or integration, such as a REST service.

See *Setting Up Process Activities*.

Milestones

Milestones represent a sub-goal within a process. Milestones are typically defined to track progress of a process.

See *Setting Up Milestones*.

Global Trigger Activity

The "global trigger activity" is an activity that appears in the process definition below the stages. It is used internally by the process to receive the incoming payload to instantiate the process instance. Updates in the transaction system flow through the global trigger activity, which in turn provide data for the conditions defined within the stages to advance the workflow. You can rename the global trigger activity and modify role assignments but otherwise you should not modify or remove it from the process definition.

> **Note:** If you remove the global trigger activity, the workflow can't be instantiated.

See *Setting Up Process Activities*.

Setting Up Connectors for Code Enforcement

In this topic we describe the process definitions connectors, or integrations, that enable Code Enforcement and OCI Automation to share data.

To access the integrations for a process definition, select **Integrations** in the left panel of the user interface. The Case Data integration combines these REST resources to enable the exchange of information about Code Enforcement cases:

- Cases (publicSectorCases)
- Case Notices (publicSectorCaseNotices)

When migrating from environment to another, such as from the Test to Production migration, make sure to update the base URL for the integration to match the URL of the new environment.

Cases

Resource Element	Value
Name	*Cases*
Resource	*publicSectorCases*
Operations	**GET** (getCases)

Resource Element	Value
Response	*BusinessData.ResponseCaseData*

Case Notices

Resource Element	Value
Name	*CaseNotices*
Resource	*publicSectorCaseNotices*
Operations	*PATCH*
Path	*{noticeKey}*
Request	*CaseNotice.CaseNotice* Parameters (*noticeKey*)
Response	*CaseNoticeResponse.CaseNoticeResponse*

Setting Up Data Storage

This topic describes how the process definition for Code Enforcement receives data and stores it for usage during the life cycle of the process instance.

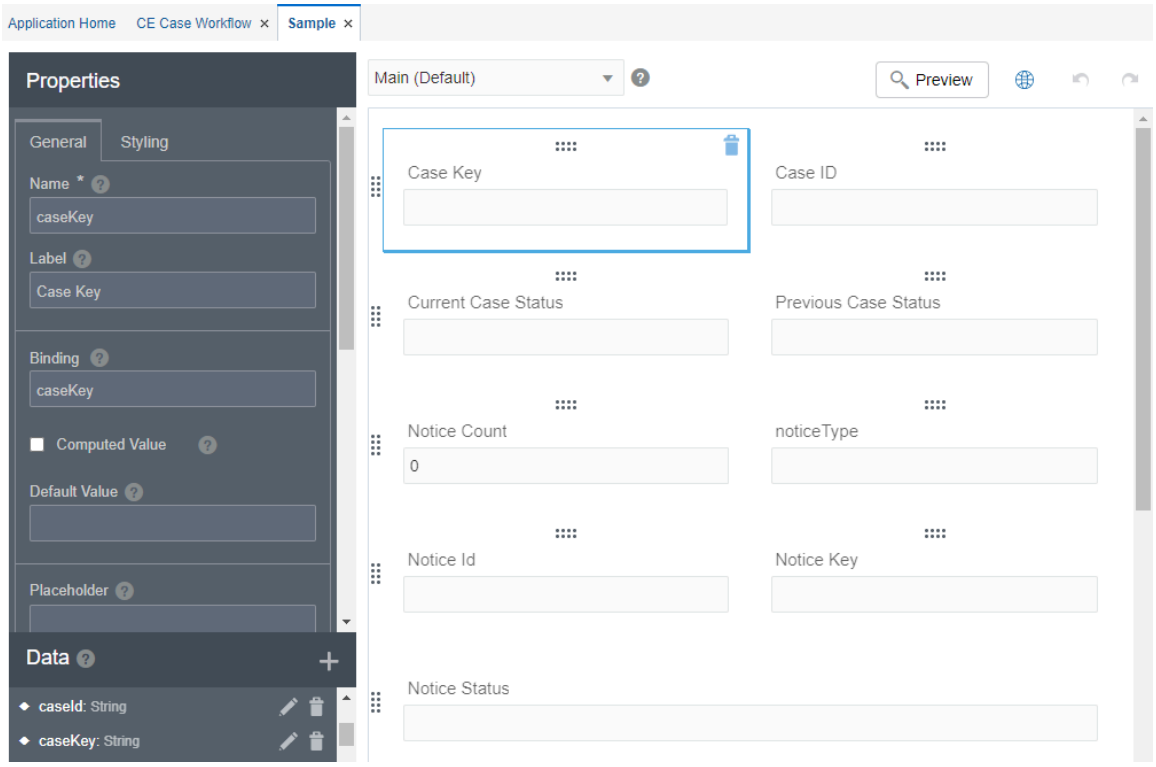
Working with Forms

When the process instance gets instantiated, the Code Enforcement system sends the required data to OCI Process Automation through a web form. The web form captures the data sent in the Code Enforcement payload.

> **Note:** Do not remove any attributes from the web form, but you can add additional attributes if your use case requires additional attributes.

Access the web form by selecting the **Forms** tab in the left panel.

This example illustrates the Forms configuration user interface.

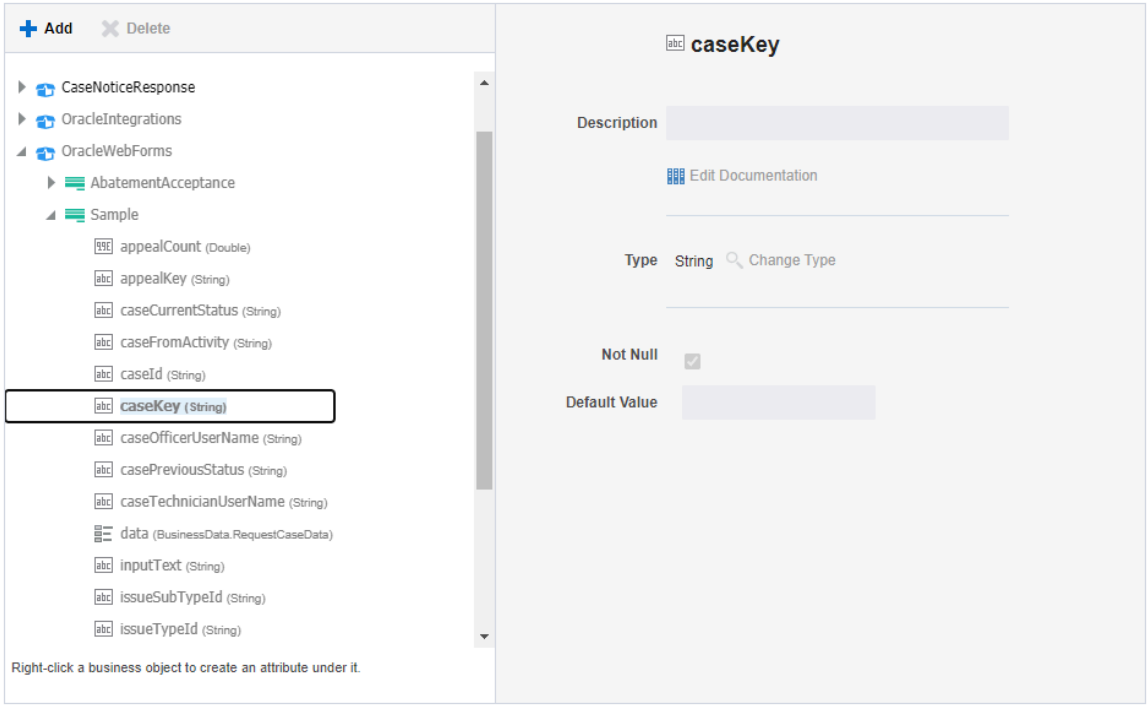


Working with Business Types

There is a one-to-one relationship between the form and the business type. The data collected through the form is then stored in the business type object. During the life cycle of the process instance, the stored data is used by the criteria you define to evaluate and carry out the workflow automation, such as activating various activities or stages when specific events occur or statuses have been set.

> **Note:** Do not remove any attributes from the business type, but you can add additional attributes if your use case requires additional attributes.

This example illustrates the Business Type configuration user interface



Setting Up Stages

This topic describes how to set up stages to group process activities and milestones.

The following topics describe the settings you can use to modify stages. For more information on stage properties, see *Define Stage Properties*.

General Settings

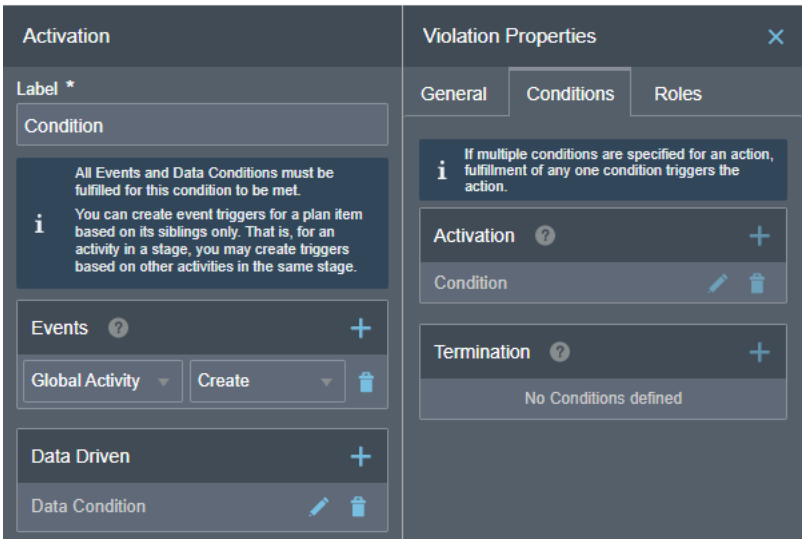
You can update:

- Name
- Description
- Markers

Conditions

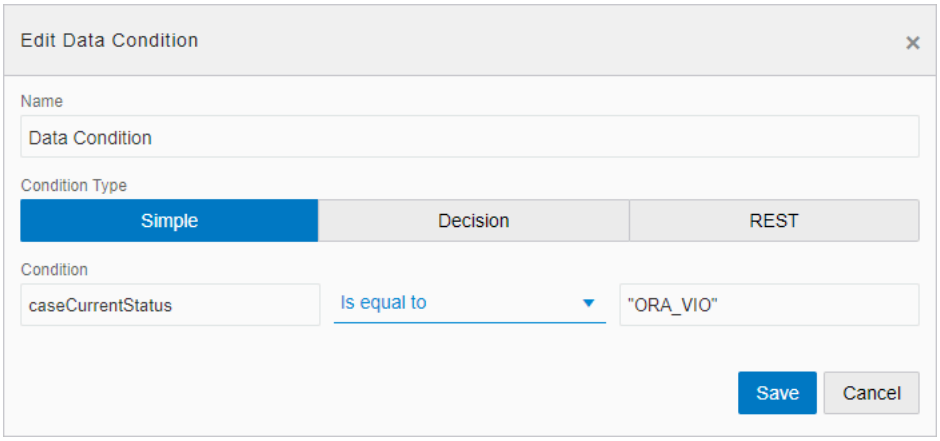
You can add additional events and conditions as needed. However, keep the initial events and conditions as is. The delivered events and conditions enable the expected activation and processing of the stages.

This example illustrates a delivered activation event and data driven condition. Details are in the surrounding text.



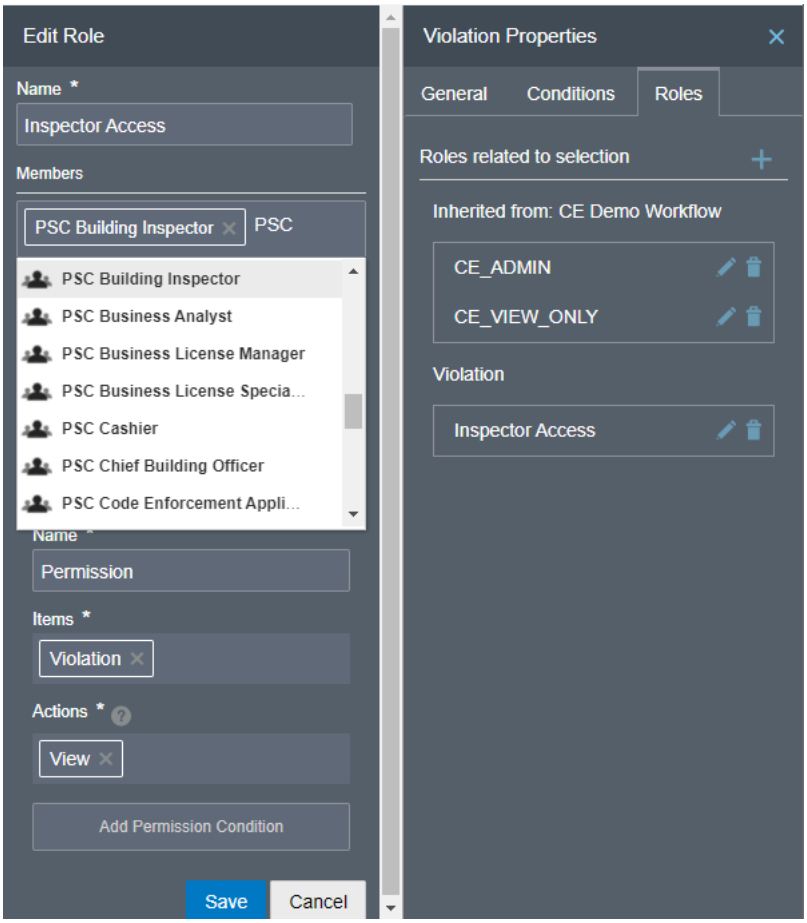
The event is based on the activation of the Global Activity, and the data driven condition is based on a specific status of an attribute in the business type. For example, for the Violation stage to activate, caseCurrentStatus must have been set to *ORA_VIO.*

This example illustrates the settings for the data driven activation condition. Details are in the surrounding text.



Roles

This example illustrates modifying roles at the stage level. Details are in the surrounding text.



On the Roles tab, notice the roles that are inherited from the top-level process definition in the **Inherited from: <Process Name>** box. Any roles you add appear below the inherited roles.

Page Element	Description
Name	Name of the role.
Members	Select PSC roles or specific users to associated with the dynamic process role. Enter partial values, such as *PSC*, to display the synchronized roles with the Code Enforcement system.
Permissions	Click **Create Permission** to add a new permission.
Name	Enter a permission name.
Items	Click in the field to display a list of items to which you can select to grant access, such as to various stages, activities, or to the entire process by selecting *Process.*
Actions	Select the access to enable for the item, such as *Update, All, None,* and so on.

The stages you define will appear at the top of the Workflow page when view case or incident details. For more information, see *Using Workflow*.

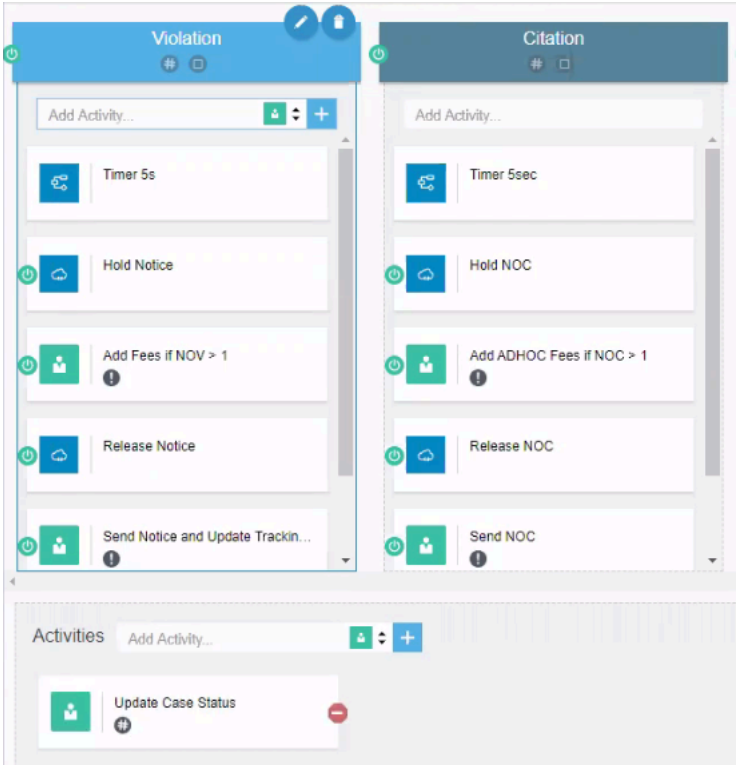
Setting Up Process Activities

This topic describes how to configure activities to reflect the human and system actions in the workflow process.

You can add as many activities as you need to meet your business requirements.

> **Note:** Do not remove the global activity at the bottom of the workspace below the stages. It is required to instantiate the process instance. You can create additional global activities if required. Global activities do not display in the run-time user interface.

This example illustrates activities within stages above the global activity at the bottom of the workspace. Details are in the surrounding text.



You can configure the General settings, Conditions settings, and Roles settings similar to stages. For more information on activity settings, see *Define Activity Properties*, and for more information on stages see *Setting Up Stages*.

> **Note:** If the first activity in a stage is a Service activity, you need to include a Process activity that calls a structured process defined with a 5 second interval (5s). In the examples, this activity is named *Timer 5s.* This creates a 5-second delay to ensure all initialization and activation processing has completed prior to starting the stage.

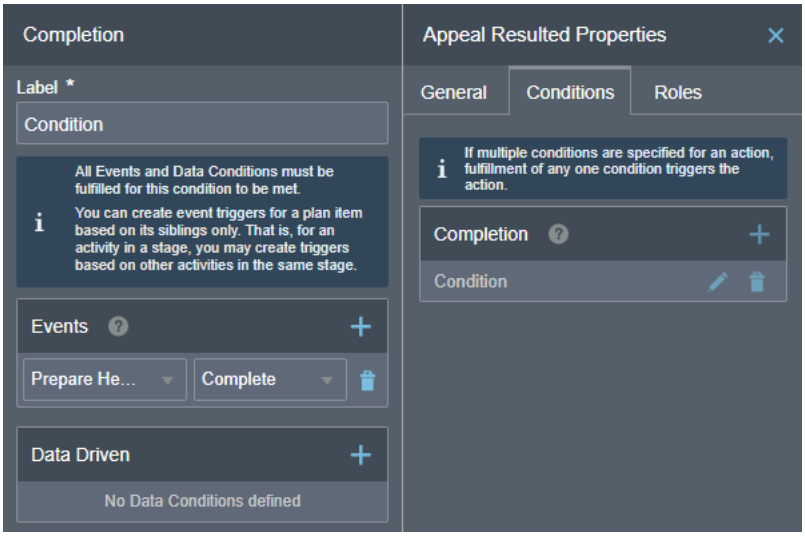
The activities defined for a stage appear on the Workflow tab when viewing case or incident details. For more information, see *Using Workflow*.

Setting Up Milestones

This topic describes how to include milestones within the stages of your process definition.

A milestone is a type of activity that represents a sub-goal within a process. Milestones are typically defined to track progress of a process. The process reaches a milestone when the status of an activity has been completed, for example.

This example illustrates a milestone. Details are in the surrounding text.



In this example, the milestone is reached when the Prepare Hearing activity has been completed.

At runtime, agency staff can view the workflow using the milestone view or the stage view. For more information, see *Using Workflow*.

Linking Process Definitions to Issue Subtypes

This topic describes how to prepare process definitions for use and how you reference them from your issue subtypes.

After you have defined your dynamic workflow process, you then need to publish and activate it so that it can be accessed and referenced by Code Enforcement. Then, you can reference it in the Code Enforcement setup pages to associate the definition with one or more issue subtypes.

Step	Link
Create a process definition group.	*Setting Up Process Definition Groups*

Step	Link
Reference the process definition group in the issue subtype.	*Setting Up Issue Subtypes*

5 Configuring Fee Decision Models

Fee Decision Model Overview

You create decision models using the Oracle Integration Cloud (OIC) decision modeling feature. Use this feature to create decision models to automate the decision logic in your business processes. As part of creating a decision model, add and order decisions, define decision inputs, and model the logic. The decision model editor supports the Decision Modeling and Notation (DMN) standard for you to create your models.

For more information on decision models refer to your OCI Process Automation documentation: *Model Decisions.*

In the Oracle Permitting and Licensing services, a decision model enables you to automate the calculation of fees based on your business process criteria.

For example, assume your agency applies varying fees based on the total cost of a building project for which a permit is being requested. A decision model enables you to automate this business logic:

- If the project value is less than or equal to $500, then the application fee is $50.
- If the project value is more than $500 but less than or equal to $1,000, then the application fee is $75.
- If the project value is more than $1,000 and $5,000, then the application fee is $125.
- For any project value over $5,000, then the application fee is $200.

Before you create a decision model, you must first create a fee item. After creating the decision models, you can then associate the decision model with a fee schedule. The fees workflow generally follows these main steps and events:

1. Create fee item(s).
2. Create decision model based on existing fee item(s).
3. Create a fee schedule incorporating fee items and decision model.
4. Associate a fee schedule with a transaction type.
5. Map intake form fields to the decision model in the Intake Form Designer.
6. When an end user is submitting an intake form the system applies fees and fee logic based on input.

Creating Decision Models for Fees

This topic describes the requirement of creating a decision model after creating fee items and before creating a fee schedule. You use OCI Process Automation to create decision models.

Prerequisites

Before you create a decision model, you need to create any required fee items that will be associated with the decision model.

For more information on fee items, see *Setting Up Fee Items*.

Configuring Decision Models

You can set up inputs and decisions any way you like following the guidelines provided in the documentation for OCI Process Automation. For more information on using OCI Process Automation to create decision models, see *Model Decisions.*.

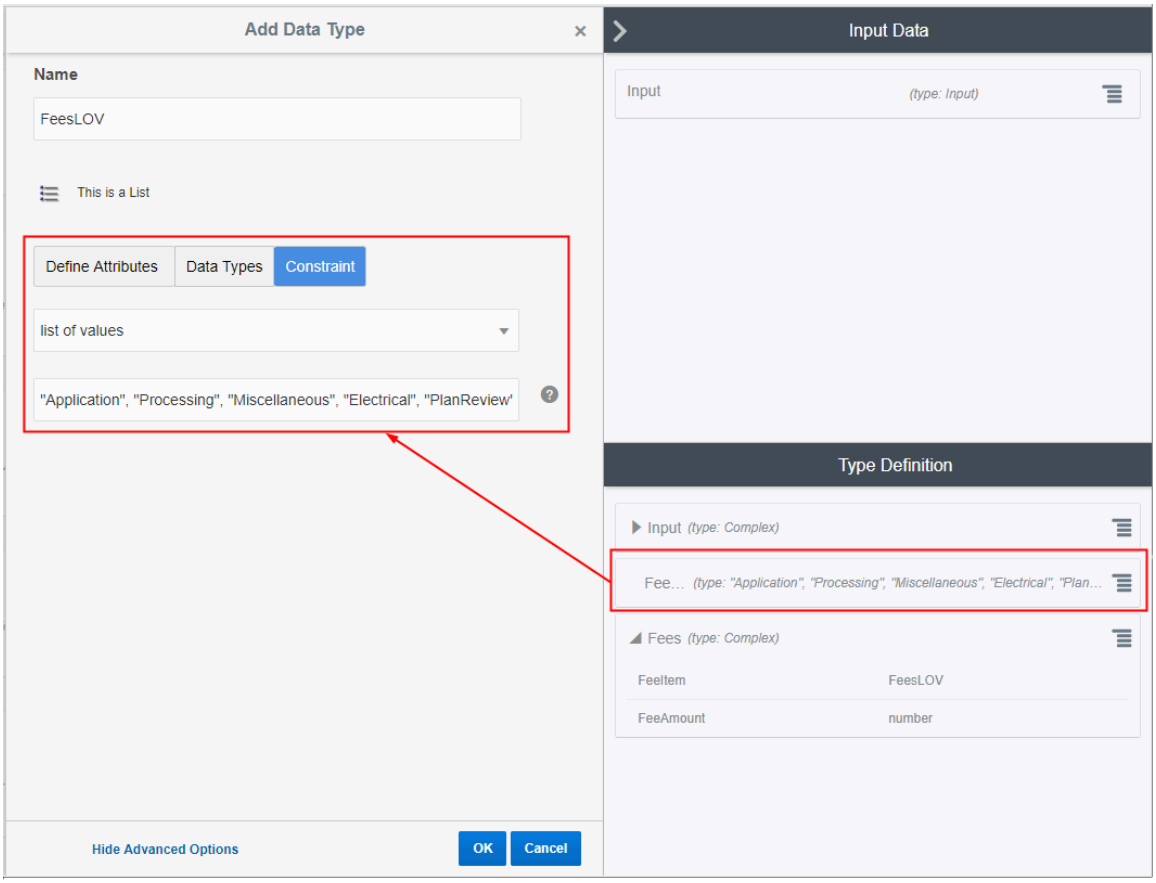
The configuration discussed here for output data types is required for setting up the interaction between the decision model and the fee schedule in Oracle Permitting and Licensing.

To configure decision models:

1. Create a fees list of values (LOV) with the fee item names.

   Currently, there is no direct integration of configuration data, such as fee items, between OCI Process Automation and Oracle Permitting and Licensing. Although creating an LOV is optional, any fee item names added to a decision model output need to be entered exactly as they appear in Oracle Permitting and Licensing. When you use the LOV and enter values in a decision output. OCI Process Automation validates the entry and displays a warning if your entry does not match an item in the LOV.

   This example illustrates the list of values used to validate fee items entered in the **Decision Table**. Select *list of values* in the **Constraint** options when you're adding the data type definition.



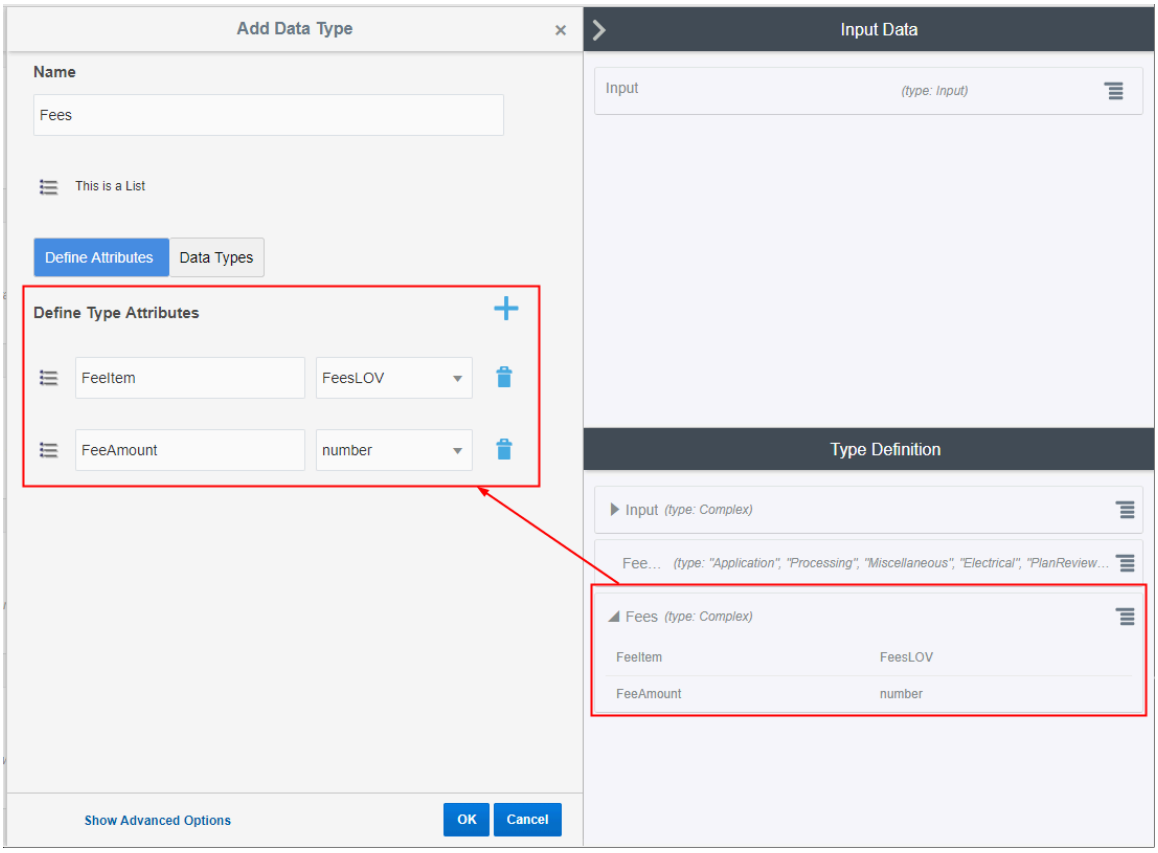
> **Note:** If the field type in the application intake form is a Check box set or a Multi-select list and the field will be used to drive fee calculations, then the fee model input should be configured as a list type. As in, select the **is List?** check box when defining the decision model input data.

2. (Required) Set up the complex data type output to include these attributes: *FeeItem* and *FeeAmount*.

   The *FeeItem* attribute should use the fees LOV that you created. The *FeeAmount* attribute uses numbers that you enter on the decision model.

   > **Note:** This step is important because the *FeeItem* and *FeeAmount* attributes are used to map the fee items on the fee schedule to the decision model.

   This example illustrates the complex data type output that you set up for your decision model. You must define *FeeItem* and *FeeAmount* attributes.

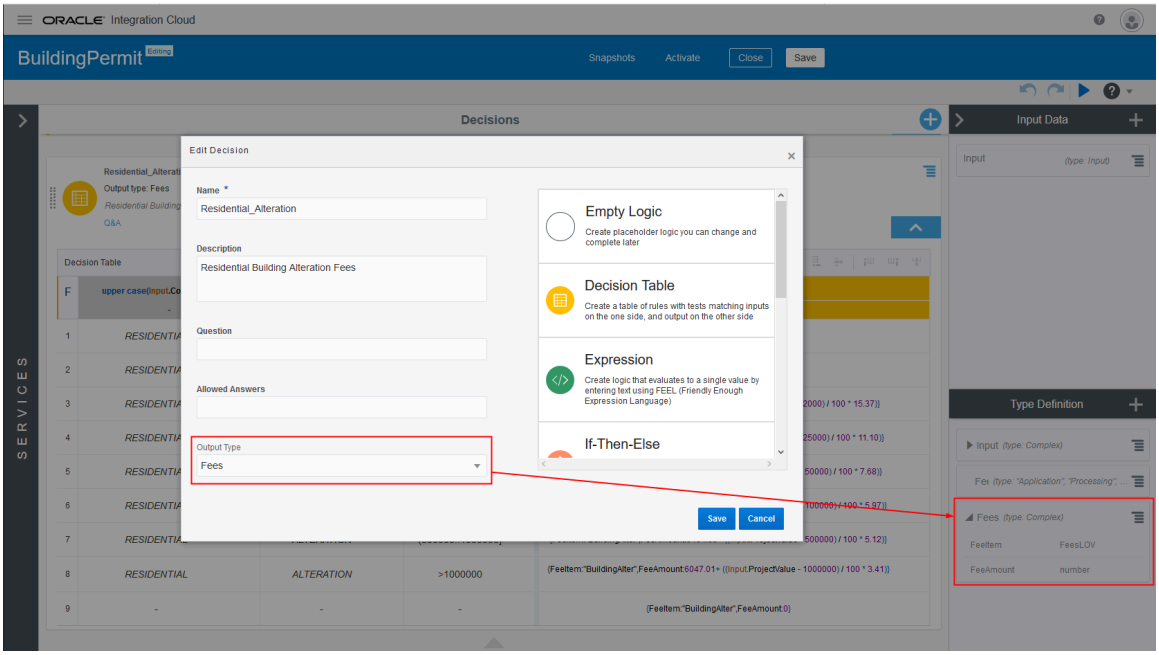


3. (Required) Add the output data type name to every decision and use allowed values in the **Decision Table** grid.

   To edit a decision, click the decision menu button and select **Edit**.

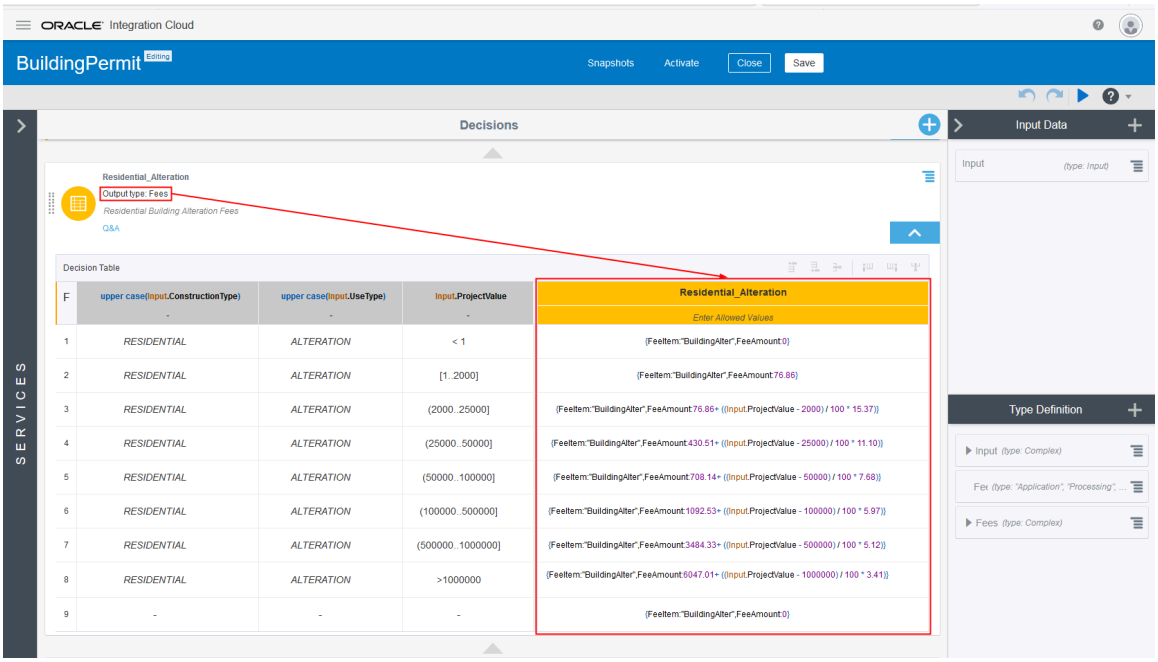
   Select the defined output type from the **Output Type** list.

   This example illustrates the output data type name *Fees* added to the Residential_Alteration decision. The output type provides constraints on which values are allowed in the decision model.



   This is an example of the final decision values based on the output type.

   The decision output type in the Residential_Alteration decision is *Fees*. The validation ran on the Fees values entered in the Decision Table, and no errors were returned.



4. (Required) You must also configure the services in the **Services** panel to the left of the decisions, and activate the model.

   For more details, see *Exposing Decisions as Services*.

When the Oracle Permitting and Licensing service submits a request to OCI Process Automation, after running the request against the decision model, the application returns the fee item name and the fee item amount.

After you have set up your decision model, you need to reference the decision model from the appropriate fee schedule. For more information on fee schedules, see *Setting Up Fee Schedules*.

Configuring Late or Recurring Fees for Code Enforcement

If the mapping source on a fee schedule line is set to either *Code Enforcement Violation Fees* or *Code Enforcement Overdue Fees*, you need to associate the fees generated by the decision model to a reference transaction, like an existing fee item. For an existing fee, you can link additional late fees to the existing fee through the associated reference transaction. When either of these mapping sources are selected, an additional field, **Map Fee Reference**, will appear, where you an select *ReferenceID,* for example.

The decision model will also need input and output attributes for ReferenceID field. The following example illustrates how you can use the ReferenceID in your calculations.

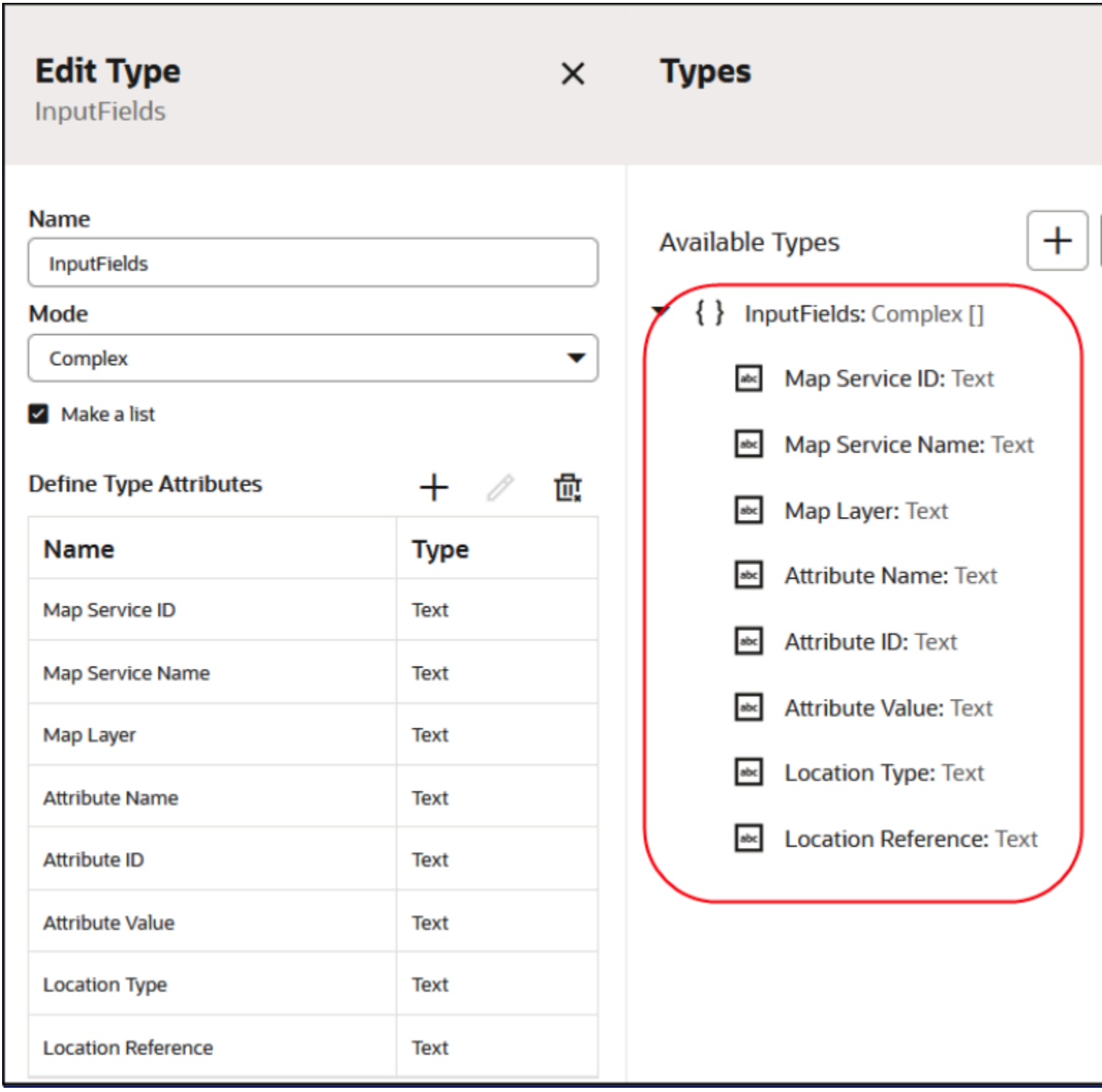
Operation	DMN Code
if	`Violation.Days Overdue >= 7`
then	`{FeeItem: "CE_RECR_FINE", FeeAmount: floor(Violation.Days Overdue / 7) * Violation.Rate * CheckPct(PenaltyPct), ReferenceID:Violation.ReferenceID}`
else	`FeeItem: "N/A", FeeAmount: 0, ReferenceID:Violation.ReferenceID}`

Configuring GIS-Based Fees

You can define fee calculation criteria using spatial data coming from the Map Layer Content field group in your intake forms. For example, agency staff can define in a decision model that a district improvement fee is required if the project location falls within a specific district.

To incorporate GIS attributes into your fee decision models, you need to:

- Include these field groups in your intake form:
    - Property *or* Business Location
    - Map Layer Content
- Create a DMN model that includes GIS-related inputs configured as a "list," which can accept multiple input values.
- Map the attributes to intake form fields as usual within the Intake Form Designer.


Edit Type
InputFields
Name
InputFields
Mode
Complex
Make a list
Define Type Attributes
Name
Type
Map Service ID
Text
Map Service Name
Text
Map Layer
Text
Attribute Name
Text
Attribute ID
Text
Attribute Value
Text
Location Type
Text
Location Reference
Text
Types
Available Types
{ } InputFields: Complex []
Map Service ID: Text
Map Service Name: Text
Map Layer: Text
Attribute Name: Text
Attribute ID: Text
Attribute Value: Text
Location Type: Text
Location Reference: Text

6 Working with Oracle Integration Cloud

Oracle Integration Cloud Overview

This topic describes how Oracle Integration Cloud (OIC) is used within Oracle Permitting and Licensing.

Oracle Integration Cloud (OIC) is used differently in Oracle Permitting and Licensing, depending on whether you are a new customer or an existing customer.

Existing customers will remain on OIC Generation 2, which is used for defining:

- Integrations
- Workflow
- Decision models (for fees)

New customers will use OCI Process Automation for workflow and decision models, while being able to license separately OIC Generation 3 for integrations. Existing customers will transition and migrate to OCI Process Automation and OIC Generation 3 in a future release.

For more information on OIC Generation 2, see *Oracle Integration Generation 2*.

For more information on OIC Generation 3, see *Oracle Integration 3*.

If using OIC Generation 3, you will need to:

- Enable OIC.

  See *Enabling Oracle Integration Cloud*.
- Create an identity domain application for OIC.

  See *Creating an Identity Domain Application for Oracle Integration Cloud*.
- Specify the identity domain application URL.

  See *Providing the Identity Domain Application URL for Oracle Integration Cloud*.
- Specify the identify domain credentials.

  See *Providing Identity Domain Credentials for Oracle Integration Cloud*.

Enabling Oracle Integration Cloud

To enable OIC:

1. Navigate to Functional Setup Manager by entering Setup and Maintenance in Page Finder.
2. Select the appropriate offering.
3. Click the Change Feature Opt In link.
4. Click the Features icon for System Administration.

5. Select Enable for Oracle Integration Cloud.
6. Click **Done**.

Creating an Identity Domain Application for Oracle Integration Cloud

Create an identity domain for OIC in IDCS.

The steps are the same for creating an identity domain for using IDCS as an identity provider.

See *Creating an Identity Domain Application for Identity Provider*.

Providing the Identity Domain Application URL for Oracle Integration Cloud

To provide an identity domain application URL for OIC:

1. In the System Administration functional area of the Functional Setup Manager, click the task: Provide Identity Domain Application URL for Oracle Integration Cloud.
2. On the Provide Identity Domain Application URL for Oracle Integration Cloud page, enter the URL for the identity domain application you created.

   You find this URL on the Identity Domain > Overview tab, in the Domain Information tab. Look for the Domain URL value. It will look similar to:

   https://idcs-12345abc.identity.xyz.<domain_name>.com

   Enter just the base URL.
3. Click **Save and Close**.

Providing Identity Domain Credentials for Oracle Integration Cloud

After you have set up an identity domain for OIC, you need to enter the domain credentials so that Permitting and Licensing can access OIC.

1. Navigate to the Identity Domain Credentials page.

   In Functional Setup Manager, select System Administration and click Provide Identity Domain Credentials for Oracle Integration Cloud.
2. On the Identity Domain Credentials page click the row for Oracle Integration Cloud.
3. Provide the following:

Page Element	Description
Client ID	The client ID of the identity domain you created in IDCS.
Client Secret	The client secret of the identity domain you created in IDCS.

4. Click **Save.**

7 Setting Up GIS

Implementing Delivered Maps

Oracle Permitting and Licensing delivers a variety of maps for use in the agency's permit, planning application, project, code enforcement, and transaction pages. There are separate maps for different users and you associate maps with a map profile based on the target audience.

For information about the delivered maps, users, navigation to maps in the system, and how the maps are used, see *Overview of Delivered Maps*.

Overview of Map Pages

This table provides information about the available map pages and their usage in Oracle Permitting and Licensing, including these offerings: Business Licenses, Code Enforcement, Permits, and Planning and Zoning.

Map Page and ID	Offerings	Navigation	Notes
Agency main map PSC_AGENCY_MAIN_MAP	All	**Navigator** > **Main Map** In the global header: **Maps (icon)** > **Main Map**	Available for agency staff.
Application intake PSC_APO	All	Start an application that includes the **Property** field group in the form design. Click the **Map View** icon to open the Property page.	This map is embedded in the Property page on the application form for address, parcel, and owner (APO) selection.
Guest/anonymous user map PSC_GUEST_MAIN_MAP	All	Click the **Explore Your City** tile on the guest landing page.	Available for guests without logging in.
Registered public user map PSC_PUBLIC_MAIN_MAP	All	Click the **Explore Your City** tile on the registered public user landing page.	Available for registered public users after logging in.
Transaction header PSC_DEFAULT_EXTENT	All	View the map in the header on the detail pages for a transaction: • Business License (not business license transactions) • Case • Inspection • Permit • Planning Application	Agencies can provide one transaction header map for everyone or separate transaction header maps for public users and agency users. The configured transaction header map applies to all offerings.
Agency - code enforcement	Code Enforcement	Click the **Code Technician Worklist** tile on the agency	Available for Code Enforcement administrators and technicians.

Map Page and ID	Offerings	Navigation	Notes
PSC_CE_AGENCY_VW_MAP		springboard then click the **Map View** icon on the worklist pages.	
Code enforcement issue intake PSC_CE_INTAKE_MAP	Code Enforcement	Select Report an Issue in the **I Want To** field and click **Go**. Or click the **Report an Issue** tile on the Code Enforcement landing page. Then select **Select an Issue Type Provide the Location of the Issue**.	Available for agency staff and registered users who have logged in as well as guests.
Mobile code enforcement inspections PSC_CE_MOBILE_MAP	Code Enforcement	Click the **Code Officer Worklist** tile on the Code Enforcement landing page. Or view the **Inspections** list page upon opening the Code Officer mobile application.	The worklist is available for Code Enforcement administrators and officers. The Code Enforcement mobile app is available for code officers.
Public - code enforcement PSC_CE_PUBLIC_VW_MAP	Code Enforcement	Click the **View Recent Issues** tile on the Code Enforcement landing page.	Available for registered public users or guests.
Permit list PSC_PERMIT_LIST	Business Licenses Permits Planning and Zoning	From the agency springboard, click a tile: **Business License Transactions**, **Permits**, or **Planning Applications**. Click the **Map View** icon on the Transactions page. From the registered public user landing page, click the **Applications** tile. Click the **Map View** icon on the Applications page.	Available for agency staff on the Transactions page and for registered public users on the Applications page.
Mobile inspection PSC_MOBILE_INSPECTION	Business Licenses Permits	Open the Oracle Inspector mobile application for permits: **Inspection Tasks** > **Inspection Detail**. Then click the **Map View** icon for an inspection.	Available for inspector supervisors and building inspectors.
Public notification PSC_PUBLIC_NOTIFICATION	Planning and Zoning	In the global header: **Maps (icon)** > **Public Notification** Or click the **Planning Applications** tile on the agency springboard and **Hearing.** Click the **Generate Notifications List** action to view the Public Notification page.	Available for agency staff.

Selecting Maps for Profiles

The available maps for a profile are filtered by the map profile users defined on the Map Profile Details page. If the profile is for *All Users,* any of the delivered maps can be selected for the profile. If the profile is for agency users or public users, a subset of maps can be selected for the profile.

> **Note:** Make sure that every map used is associated with a profile. See *Setting Up Map Profiles*. If your agency defines a new map page, the new map must be added to the Map Page lookup (ORA_PSC_COM_SYS_MAP_PROFILE) so that it is available to be associated with a map profile.

This table describes the map profile users for each of the delivered maps.

Map Page	***Agency User* Profile**	***Public User* Profile**	***All Users* Profile**
Agency main map	Yes	No	Yes
Application intake	Yes	Yes	Yes
Agency - code enforcement	Yes	No	Yes
Code enforcement issue intake	Yes	Yes	Yes
Guest/anonymous user map	No	Yes	Yes
Mobile code enforcement inspections	Yes	No	Yes
Mobile inspection	Yes	No	Yes
Permit list	Yes	Yes	Yes
Public - code enforcement	No	Yes	Yes
Public notification	Yes	No	Yes
Registered public user map	No	Yes	Yes
Transaction header	Yes	Yes	Yes

If your agency wants to display separate Transaction header maps for agency and public users, you can do this:

1. Remove the Transaction header map page set up for **All Users** from the map profile where it is used.
2. Add the Transaction header map page set up for **Public Users** to a map profile.
3. Add the Transaction header map page set up for **Agency Users** to a different map profile.

Now when you log in as agency staff, you see a map in the header of the application detail pages that's different from the map in the header that you see when you're logged in as a public user.

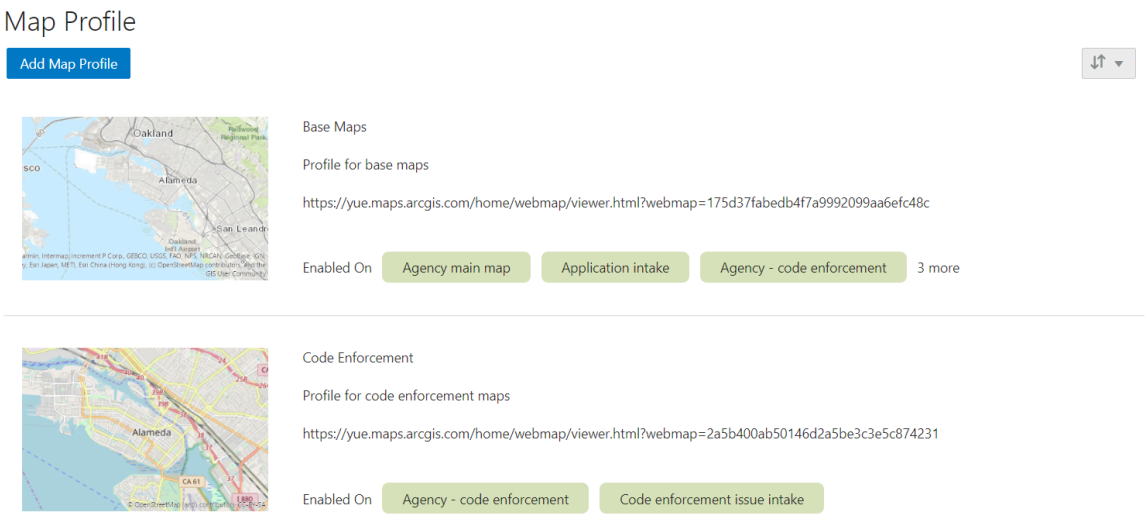
Setting Up Map Profiles

Use map profiles to configure specific instances of map functionality in the system. Profiles set the default extent of the map (the area shown by default) as well as controlling the availability of certain map options.

A profile can be linked to multiple maps. All of a profile's maps share the same map service URL, default map extent, and default base map, but the maps have individual configuration options to control end-user options for the map.

Every map must be associated with at least one map profile. In most cases, a **Switch Map Profile** button appears on the map toolbar if a map has multiple profiles, and end users can choose which profile to use. The button doesn't appear on the Transaction Header map, because you can't link to the Transaction Header map on multiple map profiles.

This example illustrates the Map Profile list page. Each row includes a thumbnail map, lists up to three maps that are associated with the profile, and states how many additional maps (if any) are associated with the profile.



Prerequisite

Mapping capabilities depend on integration with a map service such as Esri's Geographic Information Systems (GIS). Before you set up map profiles, publish your map service so that it can be referenced from within the Oracle system.

Adding a Map Profile

1. Select **GIS Setup** > **Map Profile.**

   If no profiles exist, the Map Profile Details page opens so that you can create the first profile. If at least one profile exists, the Map Profile list page appears.
2. If the Map Profile list page appears, click **Add Map Profile.**

   The Map Profile Details page appears.

3. Enter the following basic profile information:

Page Element	Description
Profile ID	Enter a unique identifier for the map profile.
Profile Name	Enter a descriptive name for the map profile. This identifies the profile on the Map Profile list page. When end users view a map with multiple profiles, they can switch profiles by choosing from a list that displays this name along with a map thumbnail.
Description	Enter a more detailed description of the map profile. This also appears on the Map Profile list page to help identify the profile.
Map Profile Users	Select the type of user who can access this map profile: - *All Users* - *Agency User* - *Public User* Only map pages that can be accessed by this type of user can be linked to the profile. For more information about which maps are available for agency, public, or all users, see *Implementing Delivered Maps*.

4. Enter the URL for your map service in the **Map Service URL** field.

   The URL to the profile provides default values for the map extent, which you will confirm or change later in this procedure.
5. If the **Base Map** field is editable, choose the type of map to display.

   The Esri server settings control whether this field is editable.

   The options are *Dark gray canvas, Light gray canvas, Imagery with labels, National Geographic, Topographic, Open Street Map, Imagery, Streets, Terrain with labels,* and *Oceans.*

   When you link specific maps to the profile, you will configure whether users can change the map type.
6. If your agency is configuring the Esri print widget, enter the URL to the print service in the **Print Service URL** field.

   If you leave this field blank, the print service URL defined on the GIS Attribute Mapping page is used.

   Proxy user setup is needed for access if the print service is secured.

7. Set the map extent.

   The map extent defines the geographical area that the map initially displays. When you create a new map profile, a generic map illustration appears above the **Choose Map Extent** option. After you choose the map extent, a preview thumbnail of your actual map extent replaces the generic illustration.

   Although the map service URL provides a default map extent, you still need to click **Choose Map Extent** to load the default extent into the profile and optionally modify it.

   > **Note:** Maps in transaction headers are initially centered on the transaction location. Therefore, the transaction header map uses the default extent from the map profile only if the transaction is not associated with a specific location.

   a. Click **Choose Map Extent.**

      The Choose Map Extent page appears. The map service URL that you previously provided sets the default map extent, and the page displays a map with that default extent.

   b. If necessary, modify the default map extent supplied by the map service URL.

      Oracle provides the ability to easily set a new map extent without making any changes to the GIS service. To change the extent, pan and zoom until you can see the desired extent, then use the **Choose Map Extent** toolbar button to draw a selection rectangle. This sets the new extent.

      The following fields describe the map extent by identifying a coordinate system and listing the minimum and maximum X and Y values on the coordinate system.

Page Element	Description
X-Min of Default Map View	The top-left X-coordinate of the initial map extent.
X-Max of Default Map View	The bottom-right X-coordinate of the initial map extent.
Y-Min of Default Map View	The bottom-left Y-coordinate of the initial map extent.
Y-Max of Default Map View	The top-right Y-coordinate of the initial map extent.
Spatial Reference	The geographic coordinate system or map projection used by the mapping service to display the map. The map service URL that you previously supplied sets this value.

   c. Click **OK** to close the Choose Map Extent window.

      The thumbnail map on the Map Profile Details page is updated to match your map extent.

8. Click **Save** to save the map profile.

Linking Maps to the Profile

To link maps to a profile and define settings:

1. Click **Add Map Page** in the Linked Map Pages section.
2. In the **Map Page** field, select a map to link to the profile.

   The maps are *Agency main map, Application intake, Agency - code enforcement, Code enforcement issue intake, Mobile code enforcement inspections, Public notification, Transaction header, Guest/anonymous map, Mobile inspection, Permit list, Registered public user map,* and *Public - code enforcement.*

   > **Note:** The maps available to link to the profile are filtered by the profile users selected on the Map Profile Details page. You can set up one transaction header map for everyone or separate transaction header maps for public users and agency users. For more information, see *Implementing Delivered Maps*.

3. Configure map-specific options.

   Depending on the map that you are configuring, some map options might not be available to enable or disable. For example, you cannot enable selection tools or window docking on the maps for mobile devices, and zoom tools are the only widgets available in the transaction header map.

   If you create any custom maps, there are no restrictions on which widgets you can enable, so take extra care when configuring those maps.

   Use these fields to configure map options:

Page Element	Description
Enable Zoom	Indicate whether the map toolbar includes **Zoom In** and **Zoom Out** tools.
Enable Default Map View	Indicate whether the map toolbar includes the **Show Default Map View** tool. This tool restores the map to its initial extent after a user zooms or pans to change the display area.
Enable Base Map Gallery	Indicate whether the map toolbar includes the **Select Base Map** tool. This tool lets users change the base map from the one specified in the profile. For example, if the profile's base map is topographic, users can change to a map with satellite imagery.
Enable Map Layers	Indicate whether the map toolbar includes the **Select Layers** tool and, depending on your GIS configuration, the **Identify GIS Information** tool. The **Select Layers** tool lets users see the list of layers and switch layer visibility on and off. Examples of layers include environmental, zoning, or infrastructure information provided by the map service. You can also turn off and turn off the sketch layer when this option is enabled. When you add the Property field group during intake form design, you can link to two separate map profiles, one for agency users and one for public users. For more information, see *Using Predefined Field Groups*.

Page Element	Description
	The **Identify GIS Information** icon gives users the ability to click map objects such as parcels to display a pop-up window with object information. This option is available if the GIS administrator has configured the GIS service to provide this information.
Enable Selection Tools	Indicate whether the map toolbar includes the **Show/Hide Selection Tools** icon. Clicking this icon opens a separate toolbar with tools for selecting and deselecting parcels on a map. The ability to select parcels on the agency main map, the registered public user map, and the guest/anonymous user map enables users to view associated transactions. Registered users and agency staff can additionally start an application for selected parcels. The ability to select parcels on the public notification map enables users to create a notification area around the selected parcels.
Enable Print	Turn on this switch to enable the print widget on this map. You can't enable the print widget on a transaction header map. You must also enter a print service URL on the map profile or on the GIS Attribute Mapping page.
Enable Sketch	Turn on this switch to enable the sketch widget on this map. You can't enable the sketch widget on a transaction header map.
Enable Detail Window Docking	Indicate whether the map detail window is docked to the side of the view. The detail window is the pop-up window that appears when a user clicks a map marker or other GIS feature such as a parcel.
Detail Window Dock Position	Specify the position where the map detail window is initially docked: *Auto, Bottom left, Bottom center, Bottom right, Top left, Top center,* or *Top right.* This field is relevant only if you enable detail window docking. This field does not apply to mobile devices, where the detail window always appears at the bottom of the screen.

4. Configure the map layer display options for the Application Intake page. The Map Layer Display Options section of the page displays all of the layers defined with the map service associated with the map profile. Administrators use this section to enable layers for GIS object selection.

> **Note:** The Map Layer Display Options section appears only for the Application Intake map page.

Page Element	Description
Unique Identifier and **Display Attributes**	Enter the identifiers and attributes that display for GIS objects when you select them on the map.

Page Element	Description
Enable Selection	When this switch is turned on for a layer, the GIS objects associated with that layer can be selected on the map.
Parcel Layer	Turn on the switch for the parcel layer. You must identify the parcel layer if you want the option to select reference parcel data independently of enabled GIS objects on the application intake map. If a parcel layer is not defined, the parcel layer on the Attribute Mapping page is used to automatically select the intersecting parcel using the map selection tools. The same is true if the parcel layer is defined and the layer is enabled. If the parcel layer is defined but the **Enable Selection** switch is turned off, intersecting parcels will not be selected when using the map selection tools. The applicant must use the Property page search box to select a parcel.

> **Note:** If you change the map service in Esri to add, update, or delete the associated layers that were enabled for GIS object selection on the map page, you need to update this setup. Delete the map page and add it again. This ensures that the latest layer information is loaded when the administrator enables layers for selection and enters attributes.

5. Click **Save** to close the Add Map Page window.
6. If necessary, click a linked map to re-open the Add Map page.
    - To edit settings, make your changes and then click **Done.**
    - To remove the map from the profile, click **Delete**.
7. Click **Save** to save the map profile.

Modifying a Map Profile

1. Select **GIS Setup** > **Map Profile**.
2. On the Map Profile list page, click the row for the profile that you want to modify.
3. Update the settings as needed.
4. Click **Save.**

Deleting Map Profiles

To delete a map profile:

1. Select **GIS Setup** > **Map Profile.**
2. On the Map Profile list page, click the row for the profile that you want to delete.
3. On the Map Profile Details page, click **Delete.**

Setting Up GIS Attribute Mapping

Use Geographic Information Systems (GIS) attribute mapping to specify property and location information about your map service parcel layer. You can also identify map layer content to collect GIS information during the application process and configure the negative buffer distance used when making selections on a map.

Prerequisites

Before you enter the information about your map service layers, you must:

- Publish the map service, which must have parcel, address, and owner layers.

  When you save the URL for a map service layer, an error message appears if the layer is not available.
- Ensure that the parcel layer has a field with parcel IDs that match the parcel IDs in the Oracle system.

  Parcel IDs must match exactly, with no formatting differences.

Setting Up the Service Layer URLs

To set up the layer service URLs:

1. Select **GIS Setup** > **Attribute Mapping.**
2. Click the **Property and Location** tab on the GIS Attribute Mapping page. Enter map service URLs in these sections:
   - Parcel Mapping
   - Address Mapping
   - Owner Mapping
   - Neighborhood Group Mapping
   - Boundary Mapping
   - Print Service
3. Enter parcel layer information:

Page Element	Description
Parcel Layer Service URL	Enter the URL for your parcel layer feature service. The URLs for the different layers of an Esri map service have numeric identifiers. The URL that you enter here ends with the number for the parcel layer. For example, `https://servername/arcgis/rest/services/Your_City/MapServer/4`. You must publish your parcel layer feature service before you enter the URL here.
Parcel Number in Parcel Layer	Select the parcel layer GIS attribute that provides the unique identifier for each parcel.

Page Element	Description
	The values in the drop-down list come from the parcel layer that you specify. Select the GIS attribute that provides the same identifiers that are used in the parcel table in the Oracle system. For information about setting up the parcel table, see *Setting Up Parcels*. On maps used as property pickers, clicking a parcel on a map retrieves the parcel identifier from the map service. This value is used as criteria for searching the Parcel table, and the search results appear in a modal window. As long as the same parcel number exists in the Parcel table, the search results include just one value, representing the selected parcel.

4. Enter address layer information:

Page Element	Description
Address Layer Service URL	Enter the URL for your address layer feature service. The URL ends with the number for the address layer. You must publish your address layer feature service before you enter the URL here.
Parcel Number in Address Layer	Select the address layer GIS attribute that provides the unique identifier for each parcel.

5. Enter owner layer information:

Page Element	Description
Owner Layer Service URL	Enter the URL for your owner layer feature service. The URL ends with the number for the owner layer. You must publish your owner layer feature service before you enter the URL here.
Parcel Number in Owner Layer	Select the owner layer GIS attribute that provides the unique identifier for each parcel.

6. Enter neighborhood layer information:

Page Element	Description
Neighborhood Group Service URL	Enter the URL for your neighborhood group layer feature service. The URL ends with the number for the neighborhood group layer. This layer identifies neighborhood groups so that the system can check whether a location on a map intersects those neighborhood groups. For example, when generating public notification

Page Element	Description
	lists, attributes are downloaded for all neighborhood groups that intersect the defined public notification area. Your GIS administrator must create and publish your neighborhood group layer feature service before you enter the URL here.

7. Enter boundary layer information:

Page Element	Description
Boundary Layer Service URL	Enter the URL for your boundary layer feature service. The URL ends with the number for the boundary layer. This layer identifies the agency's boundaries so that the system can check whether a location on a map is within those boundaries. For example, in the code enforcement system, issue locations must be within the agency's boundaries. Your GIS administrator must create and publish your boundary layer feature service before you enter the URL here.

8. Enter print service information:

Page Element	Description
Print Service URL	Enter the URL for the default print service used with print widgets on maps. This print service URL is used if the print URL isn't defined on the Map Profile for the map page.

9. Click **Save.**

Identifying Map Layer Attributes

Agencies can collect GIS information from map layer objects and save it with the application during the intake process. The agency defines a map service along with its service layers and service layer attributes on the Map Layer Content tab of the GIS Attribute Mapping page. The agency can have multiple map services, layers, and attributes. After defining the map layer content, you can select attributes to capture from all of the map services when designing the application form for a transaction type.

1. Select **GIS Setup** > **Attribute Mapping.**
2. Click the **Map Layer Content** tab on the GIS Attribute Mapping page to add one or more map services.
3. Click the **Add** button.
4. On the GIS Attributes to Capture page, define the map service:

Page Element	Description
Map Service ID	Enter a unique alphanumeric ID.

Page Element	Description
Map Service Name and **Description**	Enter a map service name and description.
Map Service URL	Enter a feature service URL or enterprise map service URL only.

5. Click **Add** define a map service layer and attribute information:

Page Elements	Description
Layer	Select a map layer identifier from the drop-down list of map layers within the map service.
Layer Label	Enter a label for the layer that the map service attributes belong to. This label identifies the map layer when you select attributes to capture during the application form design.
Attribute	Select an attribute from the drop-down list that you want to capture from the map layer attribute table within the map service.
Attribute ID	Enter an attribute ID to uniquely identify the attribute from the map service in the Oracle system.
Attribute Label	Enter a label for the GIS object attributes to be displayed when a GIS object is selected. This label identifies the attribute in a map layer when you select attributes to capture when designing an application form. Depending on the application design, this label also appears when the applicant adds a property to the intake form and in application details after submittal. **Note:** Oracle recommends entering relevant and unique attribute labels to make them distinguishable from each other while choosing attributes for a specific application form.

6. Delete and add attributes row by row using the **Add** and **Delete** actions. You can't delete an attribute row if the attribute is used on an application form.
7. Click **Save**.

For information about selecting the attributes to collect on an application form, see "Working with Property Field Groups" in *Using Predefined Field Groups*.

Configuring the Negative Buffer Distance

When you select a property on a map, the system finds all of the GIS attributes in the agency setup that intersect the selected geometry. You can use the negative buffer distance to subtract the specified distance from each side of the

geometric shape of the selected property, which helps to reduce capturing extra attribute values that may be associated with the neighboring properties. The default negative buffer distance is 2 feet.

1. Select **GIS Setup** > **Attribute Mapping**.
2. Click the **Map Layer Content** tab on the GIS Attribute Mapping page.
3. Click the **Update** (pencil) icon next to the value for the **Negative Buffer Distance**.
4. Enter a number in the **Distance** field. The default value is 2, but you can enter a different number, including 0 (zero) or a decimal value.
5. In the **Unit** field, the default unit of measure is Feet, but you can select one of the available units: Feet, Kilometers, Meters, or Miles.
6. Click **Save**.

Setting Up Access to Secure Map Services

This topic discusses how to set up access to the agency's secure private maps.

Overview of Secure Map Access

Giving users access to non-public maps (maps that can't be accessed directly through a browser) involves setup in both the Esri and Oracle systems:

1. In the Esri system, set up proxy users with access to the maps.

   You can set up proxy users with access to one or multiple secure map services.
2. In Oracle, create a secure access definition that includes the user ID and password for the proxy.

   The secure access definition also includes a URL for the map service to be accessed and a URL for the web service that generates an authentication token for accessing the map services.

On the Oracle side, create separate secure access definitions for each of the map service URLs that you need to access. You can create definitions for specific map services, such as *https://portal.city.net/arcgis/rest/services/TaxParcels/MapServer,* or for generic services such as *https://portal.city.net/arcgis/rest/services.*

When an Oracle user attempts to access a secured map, the authentication process checks first for access information for the specific map service, then for access information for the generic service.

Prerequisites

To enable access to secure map services, you must:

1. Set up proxy users in the Esri system.
2. Give these proxy users appropriate access to the secure maps services that will be accessed from the Oracle system.

Setting Up Secure Map Access

To set up secure map access:

1. Select **GIS Setup** > **Secure Map Access.**

   The GIS Secure Map Access page appears.

2. In the **Secure Map and Feature Server** section, click the **Add Proxy User** button to create a new secure access ID.
3. Enter the following information:

Page Element	Description
Secure Access ID	Enter an identifier for this configuration.
Description	Enter a description for this configuration.
GIS Service URL	Enter the URL to the secured map service that will be accessed with this configuration.
Token Service URL	Enter the URL to the web service that generates an authentication token for accessing secure map services.
User Name	Enter the user name for the proxy user that is used to access secure map services.
Password	Enter the password that the proxy user uses to access secure map services.
Confirm Password	Re-enter the password. The re-entered password is compared to the originally entered password to help catch data entry errors.

4. Click **Save.**

   Saving tests the access information that you provided. If the test fails, you aren't able to save.

Modifying a Map Profile

1. Select **GIS Setup** > **Secure Map Access.**

   The GIS Secure Map Access page appears.
2. In the **Secure Map and Feature Server** section, click the row for the proxy user access that you want to modify.
3. Update the settings as needed.
4. Click **Save.**

Deleting Map Profiles

To delete a map profile:

1. Select **GIS Setup** > **Secure Map Access.**

   The GIS Secure Map Access page appears.
2. In the **Secure Map and Feature Server** section, click the row for the proxy user access that you want to delete.
3. On the Secure Map and Feature Server detail page, click **Delete.**

8 Configuring Oracle Intelligent Advisor

Overview of Oracle Intelligent Advisor Configuration

This topic provides an overview of how Oracle Intelligent Advisor is used within Oracle Permitting and Licensing and how it is configured.

If your site already has an installation of Oracle Intelligent Advisor, you can integrate its functionality with the permits service. The policy models created in Oracle Intelligent Advisor can act as the logic models running behind questionnaires that public users fill out to determine which permits they need to apply for depending on the nature of the project they are planning.

The topics in this chapter describe the setup pages that an administrator would view and use to configure the mapping of metadata between the permits service and the Oracle Intelligent Advisor application.

> **Note:** Most of the data displayed on the pages within the Policy Modeling menu folder are read-only. In the current release, you do not create or update the definitions such as entity models, metadata models, or enumerations. These definitions are provided for your implementation in the form of delivered seed data. This documentation is provided to describe the visible information related to the policy modeling feature. Only the Metadata Models page contains fields that can be updated.

Setting Up Entity Models

This topic describes the settings used to configure entity models used when implementing Oracle Intelligent Advisor for use with the permits application.

Adding an Entity Model

> **Note:** Most of the data displayed on the pages within the Policy Modeling menu folder are read-only. In the current release, you do not create or update the definitions such as entity models, metadata models, or enumerations. These definitions are provided for your implementation in the form of delivered seed data. This documentation is provided to describe the visible information related to the policy modeling feature. Only the Metadata Models page contains fields that can be updated.

1. Select **Policy Modeling** > **Entity Models**.
2. Click **Add.**
3. On the Entity Model Details page, enter these values:

Page Elements	Description
Name	Enter a name to identify the model within the application.

Page Elements	Description
Description	Provide additional information regarding the purpose of the model.
Enabled	Use to enable or disable a model by turning the control on or off.

4. Click **Add** in the Entities grid.
5. On the Entity Details page, enter these values:

Page Elements	Description
Name	Name of the entity.
Description	Additional information to identify the entity and describe its purpose.
Hidden from Policy Modeling	If set to true, then this entry will not be present in the Get MetaData response to Oracle Intelligent Advisor.
Top-Level Entity	Indicates if the object is the highest level entity object.
Policy Modeling Name	The functional name for an entity or attribute as it appears within Oracle Intelligent Advisor.
Use as Mapped in Entity	Defines if the entity object can be selected as an input entity.
Use as Mapped Out Entity	Determines if the entity object can be selected as an output entity.
Parent Entity Name	The name of the parent entity object of a child object.
Cardinality with Parent Entity	Indicates the cardinality relationship with the parent entity object, such as one-to-one, one-to-many, many-to-one, or many-to-many.
Policy Modeling Relationship Name	The name of the relationship between two entities as it appears in Oracle Intelligent Advisor.
Supports Attachment	Determines if attachments can be collected for rows of the entity object.

6. Click **Add** in the Entity Attributes grid to add attributes for the entity.

7. On the Entity Attribute Details page, enter these values:

Page Elements	Description
Name	The system name of the entity attribute.
Data Type	The data type of the attribute as it is defined in Oracle Intelligent Advisor. For example: o java.lang.String o java.lang.Long
Primary Key	The primary key of the underlying view object.
Policy Modeling Name	The functional display name for an entity or attribute as it appears in Oracle Intelligent Advisor.
Hidden from Policy Modeling	If set to true, then this entry will not be present in the Get MetaData response to Oracle Intelligent Advisor.
Mandatory	Determines if the field *must* be mapped from an attribute in a policy model.
Policy Modeling Data Type	Describes the data type of the field defined in Oracle Intelligent Advisor. It must be specified if no enumeration-type attribute is provided, and it cannot be specified if an enumeration-type attribute is provided. Options are: o String o Boolean o Decimal o Date o Date-time o Time-of-day
Use as Mapped In Attribute	Determines if the field can mapped from an attribute for the purpose of submitting data.
Use as Mapped Out Attribute	Determines if the field can mapped from an attribute for the purpose of submitting data.
Default Value	Enter a default value for this attribute. If added, the application includes the value in the load response to Oracle Intelligent Advisor.
Enumeration Name	Specifies the ID of the enumeration that defines a field's data type.

Page Elements	Description

8. Click Save.
9. Click Save on the Entity Details page.
10. Click Save on the Entity Model Details page.

Setting Up Metadata Models

This topic describes how to set up Oracle Intelligent Advisormetadata models and define entity relationships.

Note: Most of the data displayed on the pages within the Policy Modeling menu folder are read-only. In the current release, you do not create or update the definitions such as entity models, metadata models, or enumerations. These definitions are provided for your implementation in the form of delivered seed data. This documentation is provided to describe the visible information related to the policy modeling feature. Only the Metadata Models page contains fields that can be updated.

To set up Oracle Intelligent Advisor metadata models:

1. Select **Policy Modeling** > **Metadata Models**.
2. Click Add for the Metadata Models grid.
3. On the Metadata Models Details page, add enter these values:

Note: You can update the following fields for a metadata model definition: **Supports Policy Modeling Checkpoints, Anonymous Users Can Save Data,** and **Active Model.** By default these fields are turned-off. You can turn them on according to your business requirements.

Page Element	Description
Name	Enter the functional display name of the metadata model.
Description	Provide additional description to identify the purpose of the metadata model.
Supports Policy Modeling Checkpoints	Turn on to indicate that the metadata model is designed to support checkpoints.
Anonymous Users Can Save Data	Turn on to enable the anonymous (non-signed-in user) to save data.
Active Model	Turn on to activate or deactivate the model.

4. Click Add for the Metadata Entity Relationships grid, and enter these values:

Page Element	Description
Name	Enter the entity relationship name.
Mark as Global Entity	Turn on if the entity is global.
Cardinality with Global Entity	Indicate the cardinality with the global entity (one-to-many, many-to-one, and so on).
Policy Modeling Relationship Name	The name of the relationship between two entities as it defined within in Oracle Intelligent Advisor.

5. Click Add for the Metadata Entity Links grid, and enter these values:

Page Element	Description
Source Entity Policy Modeling Name	Represents the policy modeling name for the entity in the source entity model.
Target Entity Model Name	Enter the target entity model.
Target Entity Policy Modeling Name	Represents the policy modeling name for the entity in the target entity model for this link.
Description	Provide any additional details to describe the purpose of metadata entity link.
Cardinality with Target Entity	Indicate the cardinality with the target entity (one-to-many, many-to-one, and so on).
Policy Modeling Relationship Name	The name of the relationship between two entities as it appears in Oracle Intelligent Advisor.

6. Click **Save.**
7. Click **Save** on the Metadata Entity Relationship Details page.
8. Click **Save** on the Metadata Model Details page.

Setting Up Enumerations

This topic describes how to configure enumerations for policy modeling. An *enumeration* is a tool for managing lists of potential values for a non-boolean attribute in your policy model. Enumeration are also referred to as value lists.

Note: Most of the data displayed on the pages within the Policy Modeling menu folder are read-only. In the current release, you do not create or update the definitions such as entity models, metadata models, or enumerations. These definitions are provided for your implementation in the form of delivered seed data. This documentation is provided to describe the visible information related to the policy modeling feature. Only the Metadata Models page contains fields that can be updated.

To add an enumeration:

1. Select **Policy Modeling** > **Enumerations**.
2. On the Enumerations page, click **Add.**
3. On the Enumerations Details page, enter these values:

Page Element	Description
Enumeration Name	The functional display name of the enumeration.
Enumeration Type	The data type of the enumeration, such as: o String o Number o Boolean o Time
Description	Provide additional information to help describe the purpose of the enumeration.
Child Enumeration Name	Specify the name of a linked child enumeration, as needed.

4. Click **Add** for the Enumeration Values grid.
5. On the Enumeration Value Details page, enter these values:

Page Element	Description
Enumeration Value	Enter the function display name of the enumeration value.
Description	Provide additional information to describe the purpose of the enumeration value.

6. Click **Add** for the Child Enumeration Values grid for any child enumeration values.

Page Element	Description
Child Enumeration Value	Enter the function display name of the child enumeration value.
Description	Provide additional information to describe the purpose of the child enumeration value.

Page Element	Description

7. Click **Save.**
8. Click **Save** on the Enumeration Value Details page.
9. Click **Save** on the Enumeration Details page.

Mapping Enumerations to Metadata Models

This topic describes how to map defined enumerations to existing metadata models.

> **Note:** Most of the data displayed on the pages within the Policy Modeling menu folder are read-only. In the current release, you do not create or update the definitions such as entity models, metadata models, or enumerations. These definitions are provided for your implementation in the form of delivered seed data. This documentation is provided to describe the visible information related to the policy modeling feature. Only the Metadata Models page contains fields that can be updated.

To map an enumeration to a metadata model:

1. Select **Policy Modeling** > **Enumeration Mapping.**
2. Click **Add** for the Metadata Models grid.
3. On the Metadata Model Details page, enter the name and description of the existing metadata model to which you want to map the enumeration.

   > **Note:** Once added, the remaining attributes, such as **Supports Policy Modeling Checkpoints**, **Anonymous Users Can Save Data**, and **Active Model** are read from the metadata model definition.

4. Click **Add** for the Metadata Enumeration Relationships grid.
5. On the Metadata Enumeration Relationship Details page, enter the enumeration name.
6. Click **Save**.
7. Click **Save** on the Metadata Model Details page.

Managing Proxy Users

This topic describes how to manage proxy users for enabling integration between Oracle Intelligent Advisor and your Oracle Permitting and Licensing service.

Oracle Intelligent Advisor connects to your Oracle Permitting and Licensing service through a provided web service connector named pscOpaWSConnector.

This connector requires proper WS-Security credentials to handle the transactions between the Oracle Permitting and Licensing service and the Oracle Intelligent Advisor service.

When configuring the connection within the Oracle Intelligent Advisor hub, in the WS-Security section of the New Connection page, a user ID and password is required.

The user ID entered must have the following role within their role hierarchy:

PSC Oracle Intelligent Advisor Proxy User (ORA_PSC_OPA_PROXY_USER_DUTY)

This duty role contains the following privilege:

Access Oracle Intelligent Advisor Web Service Connector Privilege (PSC_OPA_WSC_PRIV)

This privilege allows the proxy user to integrate Oracle Intelligent Advisor with your Oracle Permitting and Licensing service.

By default, the delivered SYSTEM_ADMIN has the PSC System Administrator job role, which inherits the PSC Oracle Intelligent Advisor Proxy User duty role. Any custom (cloned) role or created user must have PSC Oracle Intelligent Advisor Proxy User duty role if you intend to use that user ID as the proxy user for the Oracle Intelligent Advisor WS-Security credentials.

Managing the Oracle Intelligent Advisor Hub Endpoint

Administrators set up the Oracle Permitting and Licensing services that are required to integrate with the Oracle Intelligent Advisor using the topology entries.

To set up the Oracle Intelligent Advisor Hub endpoint:

1. Select the **Setup and Maintenance** tile on the Agency Springboard. On the Setup page, select the offering: Public Sector Permits or Public Sector Planning and Zoning and then select the OPA Questionnaire functional area.
2. On the right panel, select the task named Manage OPA Hub Endpoint to open the setup page.
3. Expand the Server Details section and fill up the following fields, with the values you received while setting up Oracle Intelligent Advisor.

Page Elements	Description
Server Protocol	Select the protocol of the Oracle Intelligent Advisor service.
External Server Host	Enter the host of the Oracle Intelligent Advisor service.
External Server Port	Enter the port of the Oracle Intelligent Advisor service.

4. Click **Save and Close**.

Managing the Oracle Intelligent Advisor Hub

Administrators set up the Oracle Intelligent Advisor hub that is required to integrate with Oracle Permitting and Licensing services.

Oracle Intelligent Advisor Setup for Integrating with Oracle Permitting and Licensing

This is a two-step process:

1. Authorizing Embedded Interviews
2. Creating Connections

Authorizing Embedded Interviews

1. Log in to the Policy Automation Hub web interface with the user credentials of *Deploy Admin*.
2. Click the Permissions tile to open the Permissions page. Click the menu in the right top of the page and select **Access Settings**.
3. On the Access Settings page, click **Add Host** under Interview Access Control.
4. In the CORS Hosts field, enter the Oracle Permitting and Licensing application host address.
5. Click **Apply**.

Creating Connections

1. Log in to the Policy Automation Hub web interface with the user credentials of *Deploy Admin*.
2. Click the **Connections** button on the banner to open the Connections page.
3. On the Connections page, click the Actions drop-down menu and select Create a new Connection option to open the New Connection page and enter values for the various fields:

Page Element	Description
Name	Enter a name for the connection.
Type	Select **Web service**.
Collection Access	Select the collection that you have created, to gain access to the connection. Click **Allow**. The default value for this field is **Default Collection**. You can include any additional collections that you want to allow access to.
URL	Enter the URL of the connector, which is deployed with other services – the FSCM base URI from the topology manager. Append the below string to the URL of the connector as shown here: <FSCM base URI>/fscmPojoService/pscOpaWSConnector?MDMN=OPAResult
Use Custom Certificate (optional)	Select to use a custom certificate defined in Policy Automation Hub. These custom certificates will be recognized by outbound *https* calls made by a Policy Automation site. If not selected, the connection will only trust the built-in root certificates.
Version	Select the following web service version:

Page Element	Description
	12.2.13
SOAP ActionPattern (optional)	Specify the *soap:operation soapAction* name expected by the web service.

OAUTH for Data Operations:

Page Element	Description
Provide OAUTH bearer token in HTTP header on Load and Save actions	Select to allow you to enter a URL parameter and enter the value *jwt* in the URL Parameter field. The token's value is passed by specifying the parameter in the query string of the interview's start session URL. This value is then passed to the Web Service connector as an **OAuth 2.0 HTTP Authorization** header whenever a Load or Save request is sent.

WS-Security:

Page Element	Description
Provide WS-Security Username token in SOAP actions.	Select the option to allow you to enter values for the fields in the section.
Applies to	Select applies to **All**.
Username	Enter a username for the purpose of connecting securely to the web service. Note that this is not related to the username of the logged-in Policy Automation Hub user. If you have installed Oracle Intelligent Advisor, then as part of the Fusion Onboarding process you must have created a user having the following Oracle Intelligent Advisor proxy user Duty role: **ORA_PSC_OPA_PROXY_USER_DUTY**. Use the same user name in this field.
New Password	Enter a password.
Include timestamp with a 5 minute expiration (optional).	Select to include a timestamp with a validity of 5 minutes. Note: The web service connector time must be synchronized to the Oracle Intelligent Advisor server.

4. Click **Save and Close** to complete the process of creating a new connection.

Managing Oracle Intelligent Advisor Policies for your Agency

This topic describes how to set up Oracle Intelligent Advisor policies for your Agency.

You can enter an Oracle Intelligent Advisor policy model at the agency level or at the offering level (for example, for the Permits offering). When the policy is used by a specific offering, the offering-specific policy model takes priority over the agency-level policy model.

> **Note:** To identify a policy model, you enter the deployment name listed in the Deployment page. The Deployment page is where the deployment and activation of policy models is managed. To access the Deployment page, log in to the Policy Automation Hub web interface with a user role of *Policy Author* or *Deploy Admin*. On the Dashboard page, click the deployments tile to open the Deployments page. From the list of all projects currently deployed, select the desired deployment name.

To define the Oracle Intelligent Advisor policies for your agency:

1. Select **Common Setup** > **Agency**.
2. Click a row on the Agency Information tab.
3. To define an agency-level policy model:
    a. Enter the policy information in the **Oracle Policy Automation ID** field on the Agency Information tab.
    b. Click **Save.**
4. To enter an offering-level policy model:
    a. Click the Features tab.
    b. Click the **Options** link for the offering you are configuring.
    c. On the Permit Options page, enter the policy information in the **Oracle Policy Automation ID** field.
    d. Click **Save.**

> **Note:** You must repeat the steps outlined in this topic and in the *Setting Up Metadata Models* topic when you are moving the content from the Test environment to your Production environment.

Purging Checkpoint Data

This topic describes how to purge checkpoint data to synchronize model definitions and maintain performance.

Periodically, it is recommend to purge the checkpoint data to maintain the integrity of the checkpoint data. When changes are made to a policy model, the stored checkpoint data can become corrupt. Also, over time, the accumulation of checkpoint data can impede performance.

To purge the current checkpoint data, administrators can run the following Enterprise Scheduler Service job: *Purge Public Sector Interview Checkpoints Job.*

This process runs a full purge for the user checkpoints for the interview specified in the Oracle Policy Automation ID field within the Agency Information page. The purge process clears all existing checkpoints for all interviews saved by all users.

To purge check point data:

1. Navigate to the Fusion Applications homepage, such as by selecting **Navigator** > **Setup and Maintenance.**
2. Select **Tools** > **Scheduled Processes.**
3. Click **Schedule New Process.**
4. Search for and select the following job: *Purge Public Sector Interview Checkpoints Job.*
5. Click **OK** and then **Submit.**

9 Setting Up Additional Integrations

Setting Up a Proxy Role and User for Interactive Voice Response

Set up a proxy user in the Oracle Security Console to give your Interactive Voice Response (IVR) system access to Oracle Permitting and Licensing.

The IVR system accesses permit information via REST, using the proxy user credentials. During a call, a public user provides the IVR system with both a permit number and their personal IVR code. The IVR system sends that code along with any request to access the permit information in the Oracle Permitting and Licensing system. Oracle Permitting and Licensing verifies that the code matches the IVR code that's stored in the permit owner's account. If the code matches, the request is honored.

In this procedure you will use the Security Console to:

1. Create a custom role for the IVR access.
2. Assign a delivered duty role to the custom role.
3. Create the IVR proxy user.
4. Assign the custom IVR role to the IVR proxy user.

For more information about using the Security Console, see: *Using the Security Console*.

Creating the IVR Custom Role

To create the PSC IVR Proxy User role:

1. Navigate to the Security Console.
   To navigate to the Security Console, you have these options:
   - In Functional Setup Manager, click the task: *Create Integrated Voice Response Proxy User.*
   - Click Setup and Maintenance on the Agency Springboard, and on the Fusion Applications home page, select **Navigator** > **Tools** > **Security Console.**
2. Select the Roles tab.
3. Click **Create Role.**
4. On the Create Role: Basic Information page enter the following:

Page Element	Value
Role Name	*PSC IVR Proxy User*
Role Code	*CUSTOM_PSCR_IVR_PROXY_USER*
Role Category	*Financials — Job Roles*

5. Click the Role Hierarchy step, and add the following duty role: *PSC Interactive Voice Recognition Proxy User (ORA_PSC_IVR_PROXY_USER_DUTY).*
6. Click the Summary step and click **Save and Close.**

Creating the IVR User

To create the IVR proxy user:

1. In the Security Console, click the Users tab.
2. On the User Accounts page, click **Add User Account.**
3. On the Add User Account page in the User Information section, enter a **Last Name** and **User Name** of your choice.

   > **Note:** The name given for the proxy user should be generic, such as *IVR Proxy User.*

4. Enter a **Password** of your choice and confirm it.
5. Click **Add Role** for the Roles grid, and assign this role to your proxy user:
   - Role Name: *PSC IVR Proxy User*
   - Role Code: *CUSTOM_PSCR_IVR_PROXY_USER*
6. Click **Save and Close.**

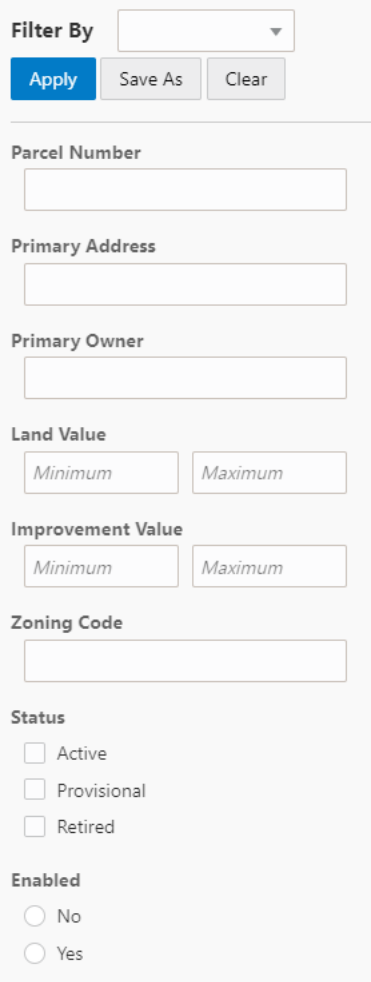
Setting Up Oracle Search Cloud Service

This topic describes how to enable Oracle Search Cloud Service (OSCS) for use with Oracle Permitting and Licensing offerings.

Using Search Cloud Service for the Parcel Page Overview

You can enable Search Cloud Service for the Parcel page. With Search Cloud Service enabled and the search index populated, when you click Filter By for the Parcel page, you can use an enhanced set of search criteria to run more flexible searches. Users can apply search criteria and save configured criteria for future use.

This example illustrates the search criteria displayed for the Parcel page when Search Cloud Service has been enabled and configured. Details are in the surrounding text.



Enabling Search Cloud Service

You opt in to the Search Cloud Service feature using the Functional Setup Manager.

To enable Search Cloud Service:

1. Access the Functional Setup Manager.
2. Select your Oracle Permitting and Licensing offering.
3. Click **Change Feature Opt in.**
4. For System Administration, click the edit icon in the Features column.
5. Select the **Enable** check box for Search Cloud Service.
6. Click **Done.**

Configuring Security

To setup access to resources required for running jobs to populate indexes, make sure that the following role has been added to the PSC System Administrator job role (ORA_PSC_SYSTEM_ADMINISTRATOR_JOB): *Application Administrator (ORA_FND_APPLICATION_ADMINISTRATOR_JOB).*

Enabling Search Framework Extensions

To enable search framework extensions:

1. From the Fusion Applications homepage, navigate to Manage Applications Core Administrator Profile Values.
2. In the **Profile Option Code** field enter *ORA_FND_SEARCH_EXT_ENABLED.*
3. In the Profile Values grid, set Site to *Yes.*

For more information on profile values, see the Oracle Applications Cloud documentation: "Profile Options" in *Implementing Applications*.

Creating the Parcel Search Index

Before you can use the Oracle Search Service features, you need to create and populate the parcel index by running an Enterprise Scheduler Service job.

To populate the search index:

1. From the Fusion Applications homepage, select **Navigator** > **Scheduled Processes.**
2. Click **Schedule New Process.**
3. On the Schedule New Process page, search for and select this process: *ESS job to create index definition and perform initial ingest to OSCS.*
4. Click **OK.**
5. On the Process Details dialog box, for the **Index Name to Reingest** field, enter *fa-psc-apo-parcel* to create and populate the parcel index.

Note: You only need to run the job to create and load the index once. After the initial run, the Search Cloud Service recognizes when new parcels have been added to the underlying view object and updates the index as needed.

Note: If you insert data directly into parcel tables using SQL scripts, for example, then for that data to be ingested into the index, you will also need to run the following job on an as-needed basis: *ESS job to run Bulk ingest to OSCS.*

10 Working With Oracle Identity Cloud Service

Overview of Identity Providers

This topic provides an overview of setting up and managing Oracle Identity Cloud Service (IDCS) as an identity provider for managing user signon and authentication.

You can use third party identity providers, such as Active Directory, or you can use Oracle Identity Cloud Service (IDCS). These topics relate to IDCS.

You will need to:

- Enable IDCS as your identity provider. See *Enabling an Oracle Identity Domain as an Identity Provider*.
- Set up an identity domain application as an identity provider. See *Providing an Identity Domain Application URL for an Identity Provider*.
- Reference the identity domain application from your Permitting and Licensing implementation. See *Providing an Identity Domain Application URL for an Identity Provider* and *Providing Identity Domain Credentials for Identity Provider*.

> **Note:** Currently, there are other steps that must be completed by the Oracle Team within your pods to ensure all internal configurations are in place. Contact Oracle Support for assistance.

When configuring Permitting and Licensing to use IDCS, consider the following:

- If you are currently using IDCS as your identity provider using the *hybridized* instance of IDCS, you just need to enable the feature in Fusion Setup Manager.
- If you are adopting IDCS as an identity provider beginning with Update 23D, you will need to perform all the steps on a *non-hybridized* instance of IDCS.

The *hybridized* instance of IDCS is the instance that is delivered with Permitting and Licensing for managing roles and authentication between the delivered components, such as OCI Process Automation for workflow.

A n*on-hybridized* instance of IDCS would be the instance you add as a separate instance to be used primarily as an identity provider.

Also, for the hybridized instance of IDCS and Permitting and Licensing, you will need to configure a similar setup for keeping roles synchronized between, for example, Security Console and IDCS. See *Creating an Identity Domain Application for Role Synchronization*.

Enabling an Oracle Identity Domain as an Identity Provider

Before you can configure an identify domain as an identity provider, you must first enable the feature in Functional Setup Manager. Once enabled, the associated setup steps will appear in the System Administration functional area in your Functional Setup Manager setup template.

To enable an Oracle Identity Domain as an Identity Provider:

1. Access Setup and Maintenance.
2. Open the appropriate setup template for your offering in Functional Setup Manager.
3. Click the *Change Feature Opt In* link.
4. For System Administration, click the *Features* link.
5. For Oracle Identity Domains as Identity Provider, select the Enable check box.

> **Note:** If you are already using the hybridized instance of IDCS as your identity provider prior to Update 23D, this is the only task you need to complete. Otherwise, you will need to create an identity domain as described in the next topic.

Creating an Identity Domain Application for Identity Provider

You will need to set up an identity domain for being an identity provider in IDCS and create a confidential application within that domain.

For information on identity domains, see the IDCS documentation here:

- *https://docs.oracle.com/en-us/iaas/Content/Identity/home.htm*
- *https://docs.oracle.com/en-us/iaas/integration/doc/adding-oracle-identity-cloud-service-identity-provider.html*
- *https://docs.oracle.com/en-us/iaas/Content/Identity/resources/manage-iam-resources.htm*

The following steps outline the main steps as an example for setting up an identity domain to be used for Permitting and Licensing. These steps need to be completed on a *non-hybridized* instance of IDCS after creating your own stripe.

To create a confidential domain application:

1. Sign in to Oracle IDCS as an administrator.
2. From the Navigation Menu, select *Identity & Security.*
3. Under Identity click Domains.
4. Click Integrated applications (or Applications, depending on your version).
5. Click **Add application.**
6. Select *Confidential Application* and click **Launch workflow.**
7. Provide a name for the application.
8. On the Configure OAuth step, select *Configure this application as a client now* under Client Configuration.
9. Under Authorization select *Client credentials.*
10. Under Token issuance policy select *Add app roles.*
11. Click **Add roles.**
12. Select and add the *User Administrator* role.

    > **Note:** All other selections aren't required and can remain with default selections.

13. In the application page:
    a. Activate the application.

b. Get the Client ID.
c. Get the Client secret.

14. Get the **Domain URL** from the identity domain Overview page.

> **Note:** You will need to provide the domain URL, client ID, and client secret values in subsequent steps.

Providing an Identity Domain Application URL for an Identity Provider

To provide an identity domain application URL:

1. In the System Administration functional area of the Functional Setup Manager, click the task: Provide Identity Domain Application URL for Identity Provider.
2. On the Provide Identity Domain Application URL for Identity Provider page, enter the URL for the identity domain application you created.

   You find this URL on the Identity Domain > Overview tab, in the Domain Information tab. Look for the Domain URL value. It will look similar to:

   https://idcs-12345abc.identity.xyz.<domain_name>.com

   Enter just the base URL.
3. Click **Save and Close**.

Providing Identity Domain Credentials for Identity Provider

After you have set up an identity domain as an identity provider, you need to enter the domain credentials so that Permitting and Licensing can access the identity provider data.

1. Navigate to the Identity Domain Credentials page.
   - In Functional Setup Manager, select System Administration and click Provide Identity Domain Credentials for Identity Provider.
   - In Permitting and Licensing, select **Common Setup** > **Identity Domain Credentials**.
2. On the Identity Domain Credentials page click the row for Identity Provider Domains.
3. Provide the following:

Page Element	Description
Client ID	The client ID of the identity domain you created in IDCS.
Client Secret	The client secret of the identity domain you created in IDCS.

Page Element	Description

4. Click **Save.**

Creating an Identity Domain Application for Role Synchronization

Note: Setting up an identity domain for IDCS role synchronization is mandatory.

Create an identity domain for role synchronization in IDCS.

These steps must be completed on the *hybridized* instance of IDCS.

The steps are the same for creating an identity domain for using IDCS as an identity provider.

See *Creating an Identity Domain Application for Identity Provider*.

Providing the Identity Domain Application URL for Role Synchronization

To provide an identity domain application URL:

1. In the System Administration functional area of the Functional Setup Manager, click the task: Provide Identity Domain Application URL for Role Synchronization.
2. On the Provide Identity Domain Application URL for Role Synchronization page, enter the URL for the identity domain application you created.

   Enter just the base URL.
3. Click **Save and Close**.

Providing Identity Domain Credentials for Role Synchronization

After you have set up an identity domain for role synchronization, you need to enter the domain credentials so that Permitting and Licensing can access the identity provider data.

1. Navigate to the Identity Domain Credentials page.
   - In Functional Setup Manager, select System Administration and click Provide Identity Domain Credentials for Role Synchronization.

- In Permitting and Licensing, select **Common Setup** > **Identity Domain Credentials**.

2. On the Identity Domain Credentials page click the row for Role Sync.
3. Provide the following:

Page Element	Description
Client ID	The client ID of the identity domain you created in IDCS.
Client Secret	The client secret of the identity domain you created in IDCS.

4. Click **Save.**

Synchronizing Roles Manually

Most of the role synchronization between Permitting and Licensing occurs automatically, but if, for example, an implementer updates role assignments directly in Security Console. In that case, that change wouldn't be reflected automatically in IDCS. As needed, it is recommended to run the Enterprise Scheduler Service job *Synchronize Agency Staff Roles.*