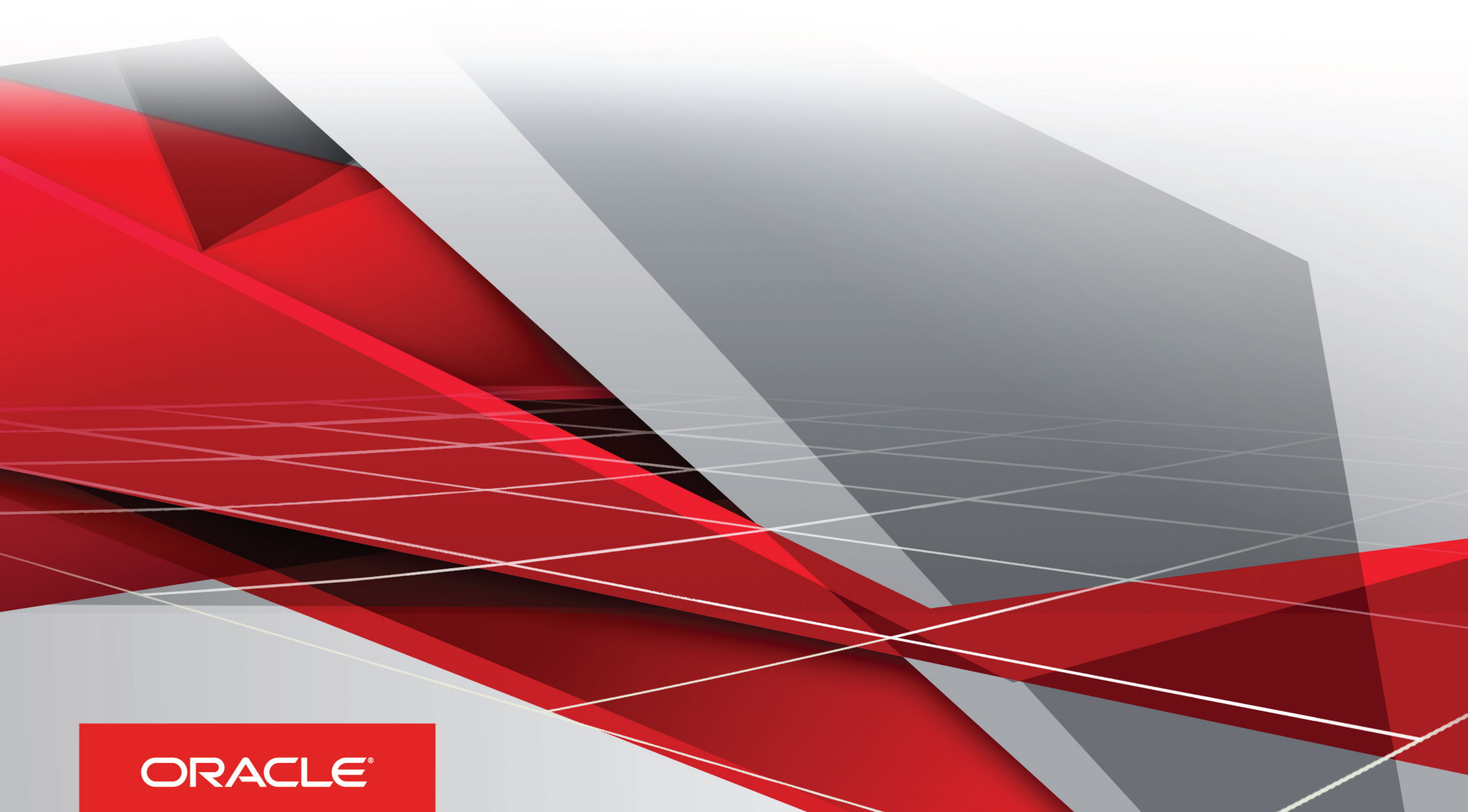
Oracle Public Sector Compliance and Regulation

Implementing Your Cloud Integrations

April 2019



Contents

Preface

This preface introduces information sources that can help you use the application and this guide.

Using Oracle Applications

This topic explains the text conventions used in this guide and points you to where you can find more information about using Oracle applications.

Conventions

The following table explains the text conventions used in this guide.

Convention	Meaning
boldface	Boldface type indicates user interface elements, navigation paths, or values you enter or select.
monospace	Monospace type indicates file, folder, and directory names, code examples, commands, and URLs.
>	Greater than symbol separates elements in a navigation path.

Additional Resources

- Community: Use *Oracle Cloud Customer Connect* to get information from experts at Oracle, the partner community, and other users.
- Guides and Videos: Go to the *Oracle Help Center* to find guides and videos.
- Training: Take courses on Oracle Cloud from *Oracle University.*

Documentation Accessibility

This topic covers accessibility concepts for this guide.

For information about Oracle's commitment to accessibility, visit the *Oracle Accessibility Program website*.

Videos included in this guide are provided as a media alternative for text-based help topics also available in this guide.

Contacting Oracle

This topic explains how to contact Oracle for support and to provide feedback.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit *My Oracle Support* or visit *Accessible Oracle Support* if you are hearing impaired.

Comments and Suggestions

Please give us feedback about Oracle Public Sector Compliance and Regulation applications help and guides! You can send an e-mail to: *PSCR_US@oracle.com*.

Oracle Public Sector Compliance and Regulation
Implementing Your Cloud Integrations

April 2019

Part Number: F12011-01

Documentation Accessibility

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

Access to Oracle Support

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

1 Setting Up Workflow

Workflow Basics

You define your permit workflow using the Process Builder in Oracle Autonomous Integration Cloud (OIC). This topic provides a general introduction to some important OIC terms and lists the high-level steps for setting up permit workflow. The list of steps includes links to additional permit-specific information. Use this permit-specific documentation in conjunction with the OIC documentation to learn how to set up workflow.

To familiarize yourself with the Process Builder in OIC. see your OIC documentation at:*https://docs.oracle.com/en/cloud/paas/integration-cloud/user-processes/develop-structured-processes.html*.

Note: Oracle provides a Solution Package with sample workflow configurations for permits. You can clone these samples and use them as starting points to create your own workflow. You can also set up your workflow from scratch.

Important OIC Terms

The OIC object where you set up your workflow tasks is called a process definition.

The following table describes the hierarchy of objects for a process definition. When you set up a permit type and you choose the appropriate process definition, you need to specify each of these hierarchical objects.

Object	Description
Space	Spaces are an organizational tool similar to a folder. Your agency chooses the spaces that make sense for your organization. For example, you can create separate spaces for different categories of permits.
Application	Applications are functional areas within Spaces. Within an application, you can access a variety of features, including processes (workflow) and integrations. 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.
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 permit type until you activate a version of the application that includes your changes. If you reuse the same version number when you activate an application, all open process instances using that version are terminated. To prevent this, use a new version number and then update any permit types that need to use the new version.
Process Definition	A process definition is a specific workflow process.

Object	Description
	When different permit 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 permit workflow.

High-Level Steps for Setting Up Workflow

The OIC documentation provides complete information on setting up workflow, but these are the high level steps, with notes about permit-specific considerations:

1. Set up a proxy role and user for accessing Oracle Integration Cloud.

   See *Setting Up a Proxy Role and User for Oracle Integration Cloud*.
2. Create the OIC space and application for your permit workflow.
3. Set up your permit-specific integrations.

   See *Setting Up the Communications Connector* and *Setting Up the Permits Connector*.
4. Create the process definition.

   Note: For permit workflow, you must create the process definition using the type Message.
5. Set up swimlanes.

   Swimlanes are equivalent to roles in the Public Sector system.
6. Design your process flow, which includes start and end events, human tasks, system tasks, gateway decision points, and arrows that define the flow through these objects.

   Your OIC documentation explains how to create a process flow, but there are additional considerations for permits. See *Setting Up Process Definitions for Workflow* for these important permit-specific tasks:
   - Setting Up Permit Data Definitions for a Process
   - Defining Arguments for the Start Event
   - Defining Data Associations for the Start Event
   - Defining Data Associations for Sending Notifications
   - Defining Data Associations for Sending Permit Status Updates
   - Defining Data Associations for Retrieving Permit Base Data
   - Defining Data Associations for Retrieving Permit Field Data
   - Defining Data Associations for Retrieving Permit Type Data
   - Defining Statuses (Outcomes) for Human Tasks
   - Defining Conditional Logic for Gateways
7. Use custom properties to add permit-specific information to the human tasks in your workflow.

   See *Using Custom Properties*.
8. Activate your application and assign it a version number.

   Activating an application makes its new and modified process definitions available to associate with a permit type. If you reuse the same version number when you activate the application, all open process instances using that version are terminated. To prevent this, use a new version number and update any impacted permit types so that they reference the new version number.

9. If this is the first time that the application has been activated, use the Manage Roles functionality in OIC to map swimlanes to roles.

   Swimlanes cannot be mapped until the application has been activated. The mapping applies to all process definitions in the application.

   See *Mapping Workflow Swimlanes to Roles*.

Managing Intersystem Connection Disruptions

There is a built-in mechanism for handling temporary unavailability of OIC. An automatic synchronization process runs every hour and scans a database table containing relevant information pertaining to any transaction, such as a permit, in the state of pending submittal. When the process discovers items in the table, it reconnects to OIC to retrieve the process instance for that transaction. After ten attempts, running once each hour, if reconnecting to OIC is not successful, the transaction becomes stale and manual intervention is required.

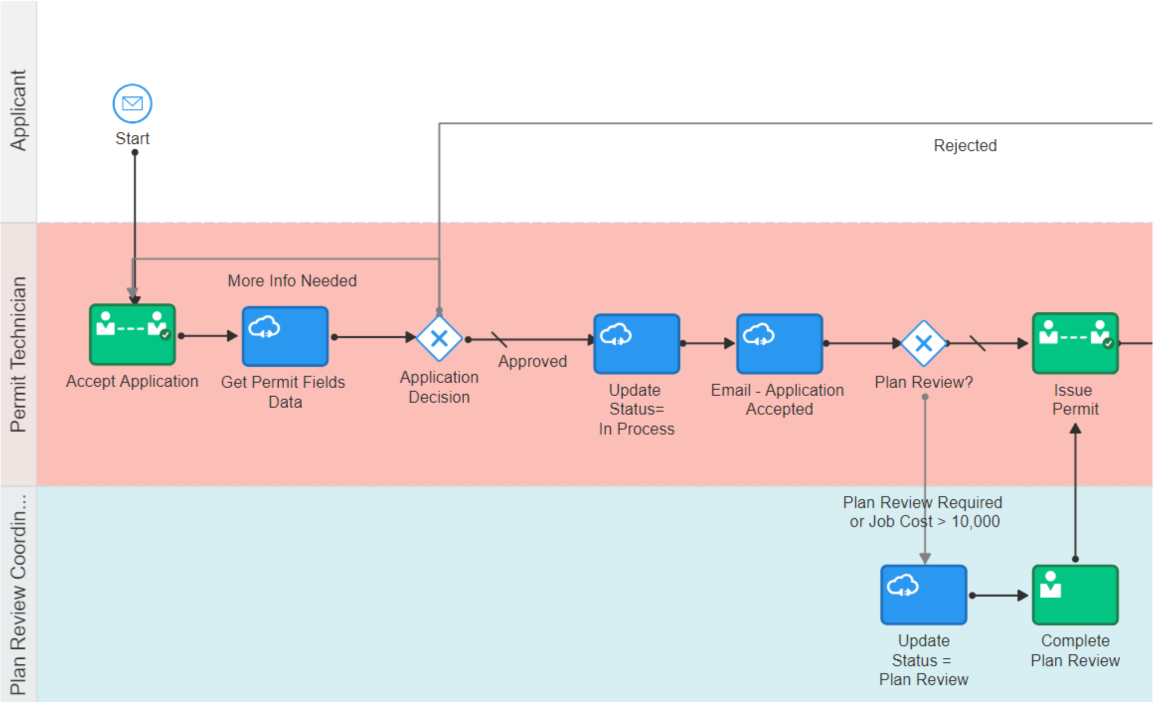
For example, if a registered user submits a permit application when OIC happens to be unavailable, that permit gets set to a state of pending submittal because no process instance can be associated with it due to OIC being unavailable. In that case, the system stores the information for that permit application in a specific database table, which will be discovered by the synchronization within an hour. Assuming OIC is available, the synchronization process retrieves the process instance from OIC, associates it with the permit application, and sets the transaction status to Submitted.

Reviewing a Sample Process Definition

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

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:

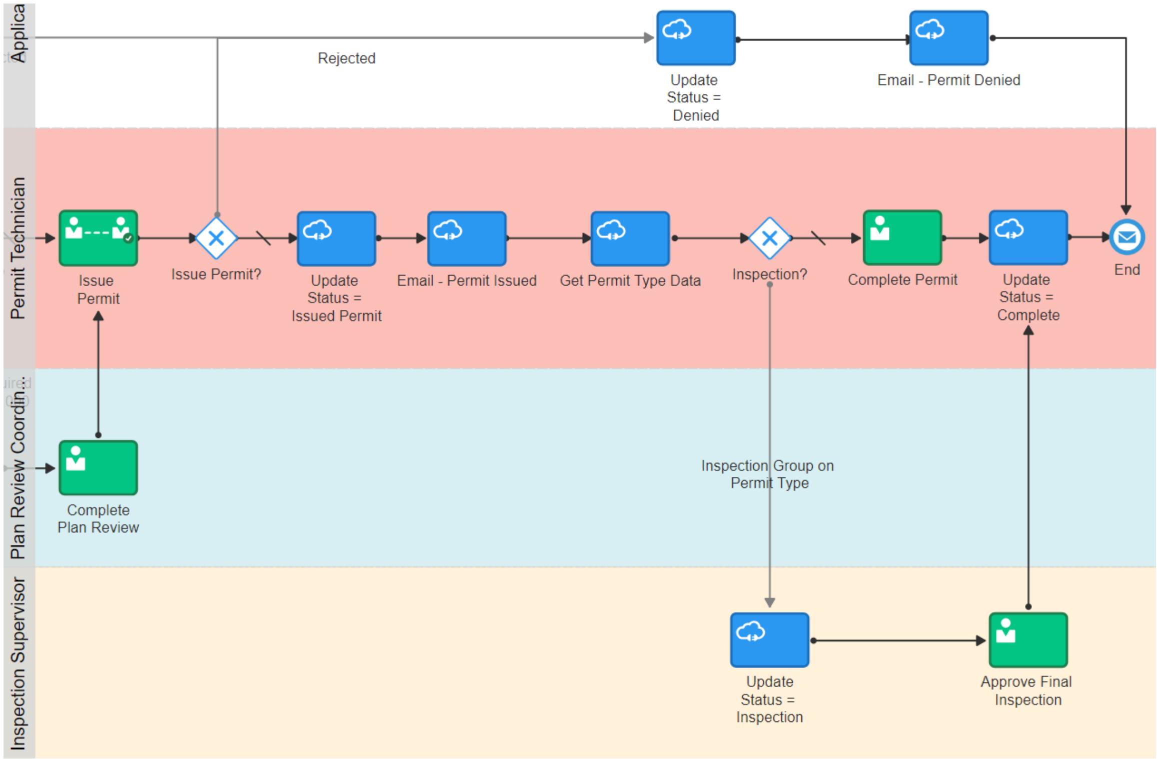
Object	Description
Swimlanes	Horizontal bands in the process map represent the roles involved in the process.
Start and End Events	All paths through the workflow process must begin at the Start event and finish at the End event.
Human tasks	Green boxes with an image of a person represent tasks that are 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 OIC 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 a Proxy Role and User for Oracle Integration Cloud

Oracle Autonomous Integration Cloud (OIC) provides the tools for setting up workflow processes. This topic provides information about using the Security Console to set up a proxy user that the Public Sector system uses to access OIC.

In this procedure, create a user and assign the PSCR Proxy User for OIC (CUSTOM_PSCR_OIC_PROXY_USER) role to that user. You use the Security Console to complete this task. This user is the OIC proxy user that the OIC system uses to connect to Public Sector Compliance and Regulation to exchange data during transaction processing.

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

To create the OIC proxy user:

1. Navigate to the Security Console.

   To navigate to the Security Console, you have these options:

   - In Functional Setup Manager, click the task: Create Process Cloud Service Proxy User.
   - 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 Use 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.
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.**

Note: You will add this proxy user to OIC process definitions.

Setting Up the Communications Connector

The communications connector enables OIC to send data to the communications center in the Oracle Public Sector 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.

Oracle provides a Solution Package with sample integration configurations. You can clone these samples and use them as starting points for your own connectors, but this procedure explains how to set up the communications connector from scratch.

The following procedure explains how to set up the communications connector with the specific integration information that is required by the Public Sector system. For general instructions related to setting up integrations in OIC, refer to your OIC documentation at *https://docs.oracle.com/en/cloud/paas/integration-cloud/user-processes/integrate-applications-and-services.html*

Note: Before you set up the communications connector, you must create the Space and the Application for your workflow processes. See *Workflow Basics*.

To set up the communication connector:

1. Go to My Oracle Support, access Doc ID 2449735.1, Public Sector Compliance and Regulation: JSON Files for Permit Integration, and download the following files that you will use later in this procedure:
   - RequestCommunications.json
   - ResponseCommunications.json
2. Access the main console in OIC.
3. In the list of OIC applications, click the application with your permit workflow.
4. Click the **Integrations** option in the left frame.
5. Click the **Create** button, then in the pop-up menu under the Create button, select **External** > **REST**
6. In the Create REST Connector window, enter the following:

Page Element	Description
Name	Enter a descriptive name such as CommunicationsConnector for the communications integration.
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
Open Immediately	Select this check box if it is not already selected.

7. Click **Create.**

   The Rest Connector Editor opens.

8. To set up security for this integration, click the **Edit** link for the Configuration section.

   Note: If you prefer to set up security when you activate the permit workflow application, you can skip the security-related steps in this procedure and skip ahead to step 13. Setting up security now simplifies the application activation steps.

9. Click the **Security** tab.
10. In the **Security Type** field, select APP Id - Basic Authentication.
11. Complete these additional fields that appear after you select the **Security Type:**

Page Element	Description
Keystore Credential	If you previously created a keystore credential, select it. Otherwise, leave this field set to [New Key] so that the system will create the keystore credential when you apply your changes.
Key Name	If you selected [New Key] as the keystore credential, enter the name to give to the new keystore. If you selected an existing keystore credential, this field is read-only and displays the key name.
Username	Enter the user name for the process cloud proxy user that you previously created. If you're using an existing keystore credential, that credential supplies a default username.
Password	Enter the password for the process cloud proxy user that you previously created. If you're using an existing keystore credential, that credential supplies a default password.

12. Click **Apply** to save the security information and close the Configuration section.
13. In the Resources section of the Rest Connector Editor, click **Add.**
14. 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.

15. In the **Operations** section, click the **Add** button and then select **POST operation** from the drop-down menu.
16. Click the new **POST** operation.
17. Enter Trigger permit communications in the **Documentation** field.

    You can leave the default values in the other fields, including leaving the **Path** field blank.
18. Click **Request**
19. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
20. Enter RequestCommunications in the **Name** field.
21. Click **Schema.**

22. ⬇ Click the **Import from File** icon next to the **Schema** button.
23. 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.
24. Click the **Import** button at the bottom of the window to save the code and close the window.
25. 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

26. Click **Response.**
27. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
28. Enter ResponseCommunications in the **Name** field.
29. Click **Schema.**
30. ⬇ Click the **Import from File** icon next to the **Schema** button.
31. 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.
32. Click the **Import** button at the bottom of the window to save the code and close the window.
33. Ensure that the following values appear for the POST operation response:

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

34. Click **Apply.**
35. Click **Save.**

Setting Up the Permits Connector

The permits connector enables OIC to exchange permit-related information with the Public Sector system.

Oracle provides a Solution Package with sample integration configurations. You can clone these samples and use them as starting points for your own connectors, but the procedures in this topic explain how to set up the permits connector from scratch.

The procedures that follow are for entering the specific information that is required by the Public Sector system. For general instructions related to setting up integrations in OIC, refer to your OIC documentation at *https://docs.oracle.com/en/cloud/paas/integration-cloud/user-processes/integrate-applications-and-services.html*

The procedures related to setting up integration connectors require you to import code using files that you download from My Oracle Support (MOS) Doc ID 2449735.1, Public Sector Compliance and Regulation: JSON Files for Permit Integration. Each procedure includes a step for downloading specific files. It's easier, however, if you download all of the files from the MOS document at once rather than returning to the MOS document multiple times.

Note: Before you set up connectors, you must create the Space and the Application for your workflow processes. See *Workflow Basics*.

Setting Up the Permits Connector

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

To set up the permits connector:

1. Access the main console in OIC.
2. In the list of OIC applications, click the application for your permit workflow.
3. Click the **Integrations** option in the left frame.
4. Click the **Create** button, then in the pop-up menu under the Create button, select **External** > **REST**
5. In the Create REST Connector window, enter the following:

Page Element	Description
Name	Enter a descriptive name such as PermitsConnector for the permits integration.
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
Open Immediately	Select this check box if it is not already selected.

6. Click **Create.**

   The Rest Connector Editor opens.
7. If you want to set up security for this integration now, click the **Edit** link for the Configuration section.

   Note: If you prefer to set up security when you activate the permit workflow application, you can skip the security-related steps in this procedure and skip ahead to step 13, where you begin setting up the outbound communications resource in this integration. Setting up security now simplifies the application activation steps.

8. Click the **Security** tab.
9. In the **Security Type** field, select APP Id - Basic Authentication.
10. Complete these additional fields that appear after you select the **Security Type:**

Page Element	Description
Keystore Credential	If you previously created a keystore credential, select it. Otherwise, leave this field set to [New Key] so that the system will create the keystore credential when you apply your changes.
Key Name	If you selected [New Key] as the keystore credential, enter the name to give to the new keystore. If you selected an existing keystore credential, this field is read-only and displays the key name.
Username	Enter the user name for the process cloud proxy user that you previously created. If you're using an existing keystore credential, that credential supplies a default username.
Password	Enter the password for the process cloud proxy user that you previously created. If you're using an existing keystore credential, that credential supplies a default password.

11. Click **Apply** to save the security information and close the Configuration section.
12. Click **Save**

Adding the PATCH Operation for Permit Statuses

✎ **Note:** Before starting this procedure, be sure to complete the procedure “Setting Up the Permits Connector.”

Permit workflow in OIC uses the PATCH operation to update the status of a permit.

To set up the PATCH operation:

1. Go to My Oracle Support, access Doc ID 2449735.1, Public Sector Compliance and Regulation: JSON Files for Permit Integration, and download the following files that you will use later in this procedure:
    - RequestPermitStatusUpdate.json
    - ResponsePermitStatusUpdate.json
2. Access the main console in OIC.
3. In the list of OIC applications, click the application for your permit workflow.
4. Click the **Integrations** option in the left frame.
5. Click the **PermitsConnector** integration.
6. In the Resources section of the Rest Connector Editor, click **Add.**
7. Expand the new Resource section that appears, and enter PermitResources in the **Name** field.
8. In the **Operations** section, click the **Add** button and then select **PATCH operation** from the drop-down menu.
9. Click the new **PATCH** operation.
10. Enter the following information:

Page Element	Value
Name	patchPermitStatus

Page Element	Value
Path	{permitResource}/{permitRecordKey} 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 permit status.

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

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

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

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

    These are example descriptions:

Parameter	Description
permitResource	Permit Resource Name
permitRecordKey	Permit Record Key

20. Click **Response.**
21. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
22. Enter ResponsePermitStatusUpdate in the **Name** field.
23. Click **Schema.**
24. Click the **Import from File** icon next to the **Schema** button.
25. Locate and upload the ResponsePermitStatusUpdate.json file that you downloaded from My Oracle Support.

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

Field	Value
Body	BusinessData.ResponsePermitStatusUpdate
Media Type	application/JSON

28. Click **Apply.**
29. Click **Save.**

Adding GET Operations for Permit Data

Note: Before starting this procedure, be sure to complete the procedure “Adding a PATCH Operation for Permit Statuses.”

In this procedure, you will set up two GET operations for fetching permit data.

- GetPermitBaseData gets general permit data that is found in all permits, such as the permit type, the permit status, and the permit applicant.
- GetPermitFieldsData gets information from the permit application (the intake form whose fields are configured using the permit designer).

To set up the GET operations for permit data:

1. Go to My Oracle Support, access Doc ID 2449735.1, Public Sector Compliance and Regulation: JSON Files for Permit Integration, and download the following files that you will use later in this procedure:
   - ResponsePermitBase.json
   - ResponsePermitFields.json
2. Access the main console in OIC.
3. In the list of OIC applications, click the application for your permit workflow.
4. Click the **Integrations** option in the left frame.
5. Click the **PermitsConnector** integration.
6. Expand the **PermitResources** resource.
7. In the **Operations** section, click the **Add** button and then select **GET operation** from the drop-down menu.
8. Click the new **GET** operation.
9. Enter the following information:

Field	Value
Name	getPermitBaseData
Path	{permitResource}/{permitRecordKey} 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 permit data, such as applicant information

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

    These are example descriptions:

Parameter	Description
permitResource	Permit Resource Name
permitRecordKey	Permit Record Key

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

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

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

20. Click **Apply.**

    This completes creation of the GetPermitBaseData operation.
21. In the **Operations** section, click the **Add** button and then select **GET operation** from the drop-down menu.
22. Click the new **GET** operation.

    The new GET operation has the default name is GetPermitResources
23. Enter the following information:

Field	Value
Name	getPermitFieldsData
Path	{permitResource}/{permitRecordKey}/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 permit data, such as job cost

24. Click **Request.**

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

    These are example descriptions:

Parameter	Description
permitResource	Permit Resource Name
permitRecordKey	Permit Record Key

26. Click **Response.**
27. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
28. Enter ResponsePermitFields in the **Name** field.
29. Click **Schema.**
30. Click the **Import from File** icon next to the **Schema** button.
31. Locate and upload the ResponsePermitFields.json file that you downloaded from My Oracle Support.

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

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

34. Click **Apply.**
35. Click **Save.**

Adding a GET Operation for Permit Type Data

Note: Before starting this procedure, be sure to complete the procedure “Adding GET Operations for Permit Data.”

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

To set up the GET operations for permit type data:

1. Go to My Oracle Support, access Doc ID 2449735.1, Public Sector Compliance and Regulation: JSON Files for Permit Integration, and download the ResponsePermitType.json file that you will use later in this procedure.
2. Access the main console in OIC.
3. In the list of OIC applications, click the application for your permit workflow.
4. Click the **Integrations** option in the left frame.
5. Click the **PermitsConnector** integration.
6. In the header of the Resources section, click **Add** to create a new permit type resource.
7. Expand the new Resource section that appears, and enter the following information:

Field	Value
Name	PermitTypeResource
Path	publicSectorRecordTypes

8. In the **Operations** section, click the **Add** button and then select **GET operation** from the drop-down menu.
9. Click the new **GET** operation.
10. Enter the following information:

Page Element	Value
Name	getPermitTypeData
Path	{permitResource}
Documentation	Get permit type setup data

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

    Here is an example description:

Parameter	Description
permitResource	Permit Resource Name

13. Click **Response.**
14. Click the **+** icon next to the **Body** field to open the Import Business Object from JSON window.
15. Enter ResponsePermitType in the **Name** field.
16. Click **Schema.**
17. Click the **Import from File** icon next to the **Schema** button.
18. Locate and upload the ResponsePermitType.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 GET operation response:

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

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

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: The procedures in this topic relate to the specific requirements of permit workflow. To create your permit workflow, you first need to become familiar with OIC and, in particular, the process builder in OIC. For more information, refer to your OIC documentation at *https://docs.oracle.com/en/cloud/paas/integration-cloud/user-processes/develop-structured-processes.html*

Setting Up Permit Data Definitions for a Process

Data definitions provide a structure for storing data from the permit system. Every process definition that you create needs the same data definitions, including:

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

  The data definition for permit fields includes all possible fields on the permit intake form, even though permits use only the subset of fields that are appropriate for the type of permit. Any fields that are not part of a specific permit remain blank when the workflow process retrieves the permit field data.

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

Note: Before you set up your data definitions, you need to set up the permit connector for the application. This is because the permit connector's three GET operations have the schema for the data. Setting up the permit connector is described in the topic *Setting Up the Permits Connector*.

To set up data definitions in a process definition:

1. Access the process definition in OIC.
2. Click **Data Objects.**
3. Set up the data definition for permit 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	permitBaseData
Data Type	Business
The drop-down list for data types	BusinessData.ResponsePermitBase

   c. Click **Create** to create the data definition and return to the Data Objects window.
4. Set up the data definition for permit field data:

a. Click **Add.**
b. Enter the following information:

Page Element	Value
Name	permitFieldsData
Data Type	Business
The drop-down list for data types	BusinessData.ResponsePermitFields

c. Click **Create.**

5. Set up the data definition for permit type data:

a. Click **Add.**
b. Enter the following information:

Page Element	Value
Name	permitTypeData
Data Type	Business
The drop-down list for data types	BusinessData.ResponsePermitType

c. Click **Create.**

6. Create simple string data objects for the permit fields that contain identifying information about permits and permit types:

To create these strings:

a. Click **Add.**
b. Set up the string using the values in this table, where each row represents a separate data definition that you need to create:

Name	Data Type	Drop-down list for the data type
recordKey	Simple	String
recordTypeKey	Simple	String
ExternalBaseURL	Simple	String
resourceName	Simple	String

c. Click **Create.**
d. Repeat for all additional rows in the table.

7. Click **Close** to close the Data Objects window.
8. Click **Save.**

Defining Arguments for the Start Event

When a permit instantiates its workflow process, it passes parameters such as the permit ID to the workflow system. The Start event must have arguments for these parameters.

To set up the arguments for the start event:

1. Open the process definition and select the Start event.
2. Open the event 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. Add the following rows to the **Arguments Definition.**

Name	Type
RecordKey	String
RecordTypeKey	String
ExternalBaseURL	String
ResourceName	String

6. 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 permit..

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 input data associations.

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
RecordKey	recordKey	The permit ID.
RecordTypeKey	recordTypeKey	The permit type ID.
ExternalBaseURL	url	The URL for the permits system.
ResourceName	resourceName	The name of the REST API resource for permits.

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

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 Events*.

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 Center in the public sector system. The source data string must be in quotation marks, and it must exactly match the identifier of an event. Oracle delivers five communication events, LNP_WORKFLOW_001 through LNP_WORKFLOW_005.
[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 permit application.
"LnpRecordKey"	body.recordFirstKeyName	The name of the key field for permits
recordKey	body.recordFirstKeyValue	The permit 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.

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
true or false	body.notification	This Boolean field indicates whether the notification is sent as an in-product notification. . 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 Permit Status Updates

The data associations for a permit status update task define the information that the task sends to the permit system.

To set up the data associations:

1. Access the process definition and select the system task that updates the permit 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
resourceName	resourceName	The unique system identifier for the permit type.
recordKey	recordKey	The permit ID
[new permit status] For example: "Accepted"	body.status	The status to be assigned to the permit. The source data string must be in quotation marks, and it must exactly match one of the valid statuses in the permit application.

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

Defining Data Associations for Retrieving Permit Base Data

The data associations for a task that retrieves permit 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 permit 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
resourceName	permitResource	The unique system identifier for the permit type.
recordKey	permitRecordKey	The permit 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	permitBaseData	This business object contains all of the permit base data. Mapping individual fields would be much more complex and is not necessary.

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

Defining Data Associations for Retrieving Permit Field Data

The data associations for a task that retrieves permit 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 permit 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
resourceName	permitResource	The unique system identifier for the permit type.
recordKey	permitRecordKey	The permit 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	permitFieldsData	This business object contains all of the permit fields. This includes all fields that

Source Data (Left side of the map)	Target Data (Right side of the map)	Description
		can be included on the permit intake form, whether or not the field exists for a specific permit. 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 permit data. Mapping individual fields would be much more complex and is not necessary.

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

Defining Data Associations for Retrieving Permit Type Data

The data associations for a task that retrieves permit 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 permit 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
resourceName	permitResource	The unique system identifier for the permit 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	permitTypeData	This business object contains all permit 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 **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, OIC 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. Permit field data (the data from the intake form) is nested within the items element under PermitFieldsData.

   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.

Using Custom Properties

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

These custom properties are available for permit workflow and are described in more detail below:

Property	Usage	Values
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.	Integers

Property	Usage	Values
PSC_FINAL_ACTIVITY	Use this property to identify a human task that is not allowed to progress when the permit has a condition that applies the **Prevent Issue or Final** business rule. In particular, use this property to identify the final human task in the process definition. See *Setting Up Conditions* and *Applying Conditions to Applications*.	Yes identifies the final activity. A blank value or a No value means that the task is not the final activity.
PSC_ACTIVITY_TYPE	Use this property to identify the final inspection task in the process. Setting this property is necessary to support the permit 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.
PSC_AUTO_UPDATE_ACTION	Use this property to identify the action to take when auto-advancing the final inspection task in a process. Setting this property is necessary to support the permit business logic that auto-advances the inspection task when the last inspection is closed.	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.

Making Custom Properties Available in a Process Definition

Before you can use a custom property, you need to add the property to the process definition. You must do this for each of your process definitions.

To add a custom property to a process definition:

1. Access the process definition and click the **#** (Custom Properties) toolbar icon.
2. Enter the following values in the **Property Name** and **Description** fields in the Custom Properties list:

   Note: You must use the exact property names given in this procedure. You can, however, alter the descriptions.

Property Name	Description
PSC_LIST_ORDER	Human tasks display order
PSC_FINAL_ACTIVITY	Identify the final human task
PSC_ACTIVITY_TYPE	Identify the type of activity such as inspection
PSC_AUTO_UPDATE_ACTION	Identify the action to take when auto-advancing

3. Click **OK.**

4. Click **Save.**

PSC_LIST_ORDER Property

The Workflow page for a permit 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.

To assign order numbers to human tasks:

1. Analyze all human tasks in the process and decide on the appropriate order.

   Tasks appear in ascending numerical order. You assign order numbers one task at a time, so if you later decide to change the order, you have to update each affected task individually.
2. Access the process definition.
3. Select a human task.
4. Open the task properties.
5. Select the **Business Properties** icon in the icon bar, then select **Custom Properties** from the list of options that are associated with that icon.
6. Enter a number in the **PSC_LIST_ORDER** custom property.
7. Close the Properties window and save.
8. Repeat the previous steps for all human tasks in the process definition.
9. Click **Save.**

PSC_FINAL_ACTIVITY Property

Human tasks that you identify as a final activity cannot advance when a permit has a condition that applies the **Prevent Issue or Final** business rule.

To identify the final human task using the PSC_FINAL_ACTIVITY property:

1. Access the process definition.
2. Select the final human task.
3. Open the task properties.
4. Select the **Business Properties** icon in the icon bar, then select **Custom Properties** from the list of options that are associated with that icon.
5. Enter Yes in the **PSC_FINAL_ACTIVITY** custom property.

   To remove a Yes value from this property, set the value to No or clear the value and leave it blank.
6. Close the Properties window and save.
7. Click **Save.**

PSC_ACTIVITY_TYPE and PSC_AUTO_UPDATE_ACTION Properties

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.

To set up custom properties for auto-advancing an inspection task:

1. Access the process definition.
2. Select the human task that represents final inspections.
3. Open the task properties.
4. Select the **Business Properties** icon in the icon bar, then select **Custom Properties** from the list of options that are associated with that icon.
5. Enter Inspection in the **PSC_ACTIVITY_TYPE** custom property.
6. Enter the desired action in the **PSC_AUTO_UPDATE_ACTION** custom property.

   The action you enter is the action to be taken when the task is successfully complete—that is, when the permit passes its final inspection. Take care to use the correct spelling, capitalization, and punctuation for the action name.
7. Close the Properties window and save.
8. Click **Save.**

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. After the OIC application containing the process definition is activated, use the Manage Roles functionality in OIC to map the swimlanes to roles. The mapping applies to all process definitions in the OIC application.

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.

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 needs to be mapped to the OIC proxy user role, or to the OIC proxy user that you created in the following: procedure *Setting Up a Proxy Role and User for Oracle Integration Cloud*. In the following instructions, you will map the swimlane to the role.

To map swimlanes to roles:

1. Access the My Tasks area of Oracle Autonomous Integration Cloud.
2. Click the **Workspace** button in the right frame.
3. Click **Administration** in the left frame.
4. If the Manage Roles page does not appear by default, click **Manage Roles** in the left frame.

   The Manage Roles page lists process roles using the format [application].[swimlane].
5. Search for your application to filter the list.

   The **Process Owner** and **Process Reviewer** roles are part of all applications. Other swimlanes in the list are ones that you created in your process definitions.
6. Add the delivered role PSCR Proxy User for OIC to the swimlane with the Start event:
   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.

b. In the **Assign Roles** list for the selected swimlane, click **Add Member**.

c. In the dialog box for adding members, search by Groups for PSCR Proxy User for OIC.

   A group in OIC is equivalent to a role in the Public Sector system.

d. In the search results, select PSCR Proxy User for OIC and then click **OK** to assign the role to the swimlane and return to the list of swimlanes.

2 Configuring Fee Decision Models

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 Oracle Autonomous Integration Cloud 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*.

Decision Model Overview

You create decision models using the Oracle Autonomous 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 Oracle Autonomous Integration Cloud documentation, *Create Decisions*.

In the Public Sector Compliance and Regulation 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 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.

Configuring Decision Models

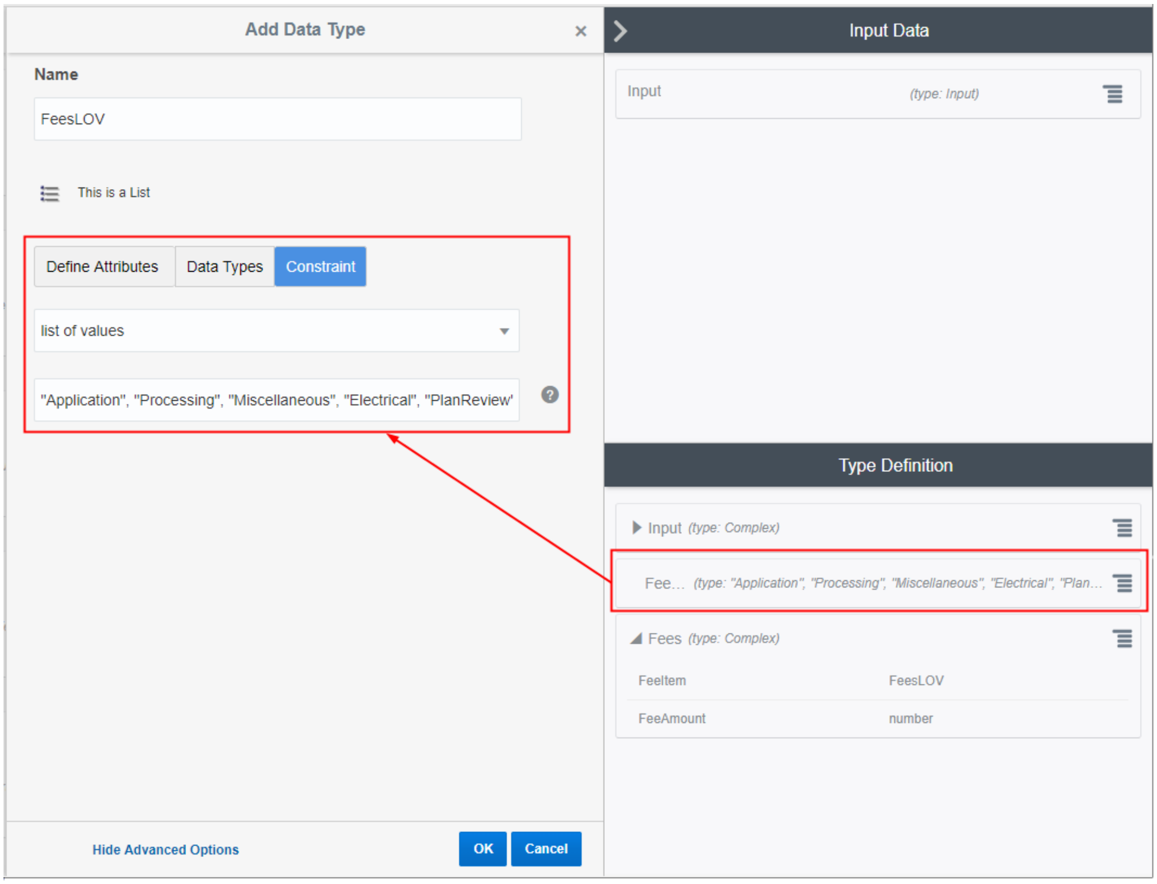
You can set up inputs and decisions any way you like following the guidelines provided in the documentation for OIC. However, the configuration here for output data types is required for setting up the fee schedule.

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 OIC and Oracle Public Sector Compliance and Regulation. 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 Public Sector Compliance and Regulation. When you use the LOV and enter values in a decision output. OIC 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 setup in OIC 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.



Add Data Type

Name: FeesLOV

This is a List

Define Attributes | Data Types | Constraint

list of values

"Application", "Processing", "Miscellaneous", "Electrical", "PlanReview'

Input Data

Input (type: Input)

Type Definition

Input (type: Complex)

Fee… (type: "Application", "Processing", "Miscellaneous", "Electrical", "Plan…

Fees (type: Complex)

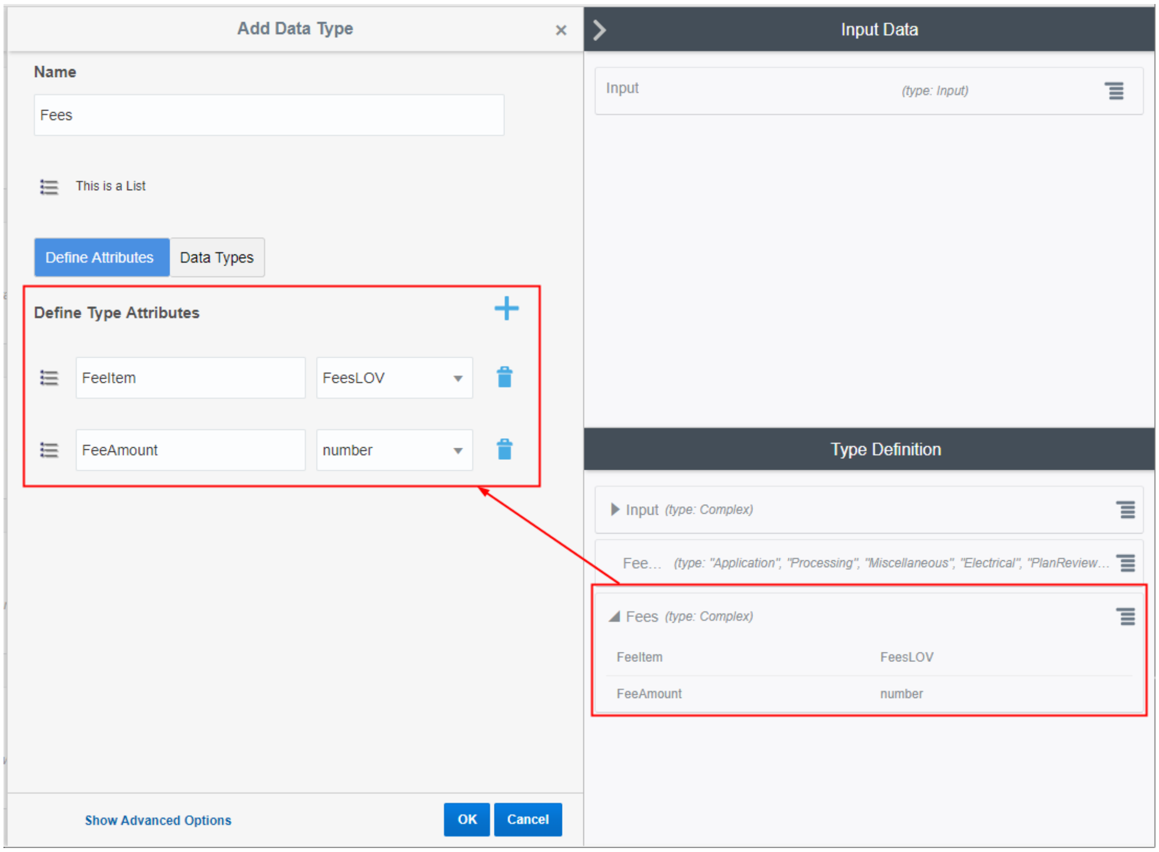
FeeItem	FeesLOV
FeeAmount	number

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 in OIC for your decision model. You must define FeeItem and FeeAmount attributes.



Add Data Type

Name: Fees

This is a List

Define Attributes | Data Types

Define Type Attributes

FeeItem	FeesLOV
FeeAmount	number

Show Advanced Options

OK | Cancel

Input Data

Input (type: Input)

Type Definition

Input (type: Complex)

Fee… (type: "Application", "Processing", "Miscellaneous", "Electrical", "PlanReview…

Fees (type: Complex)

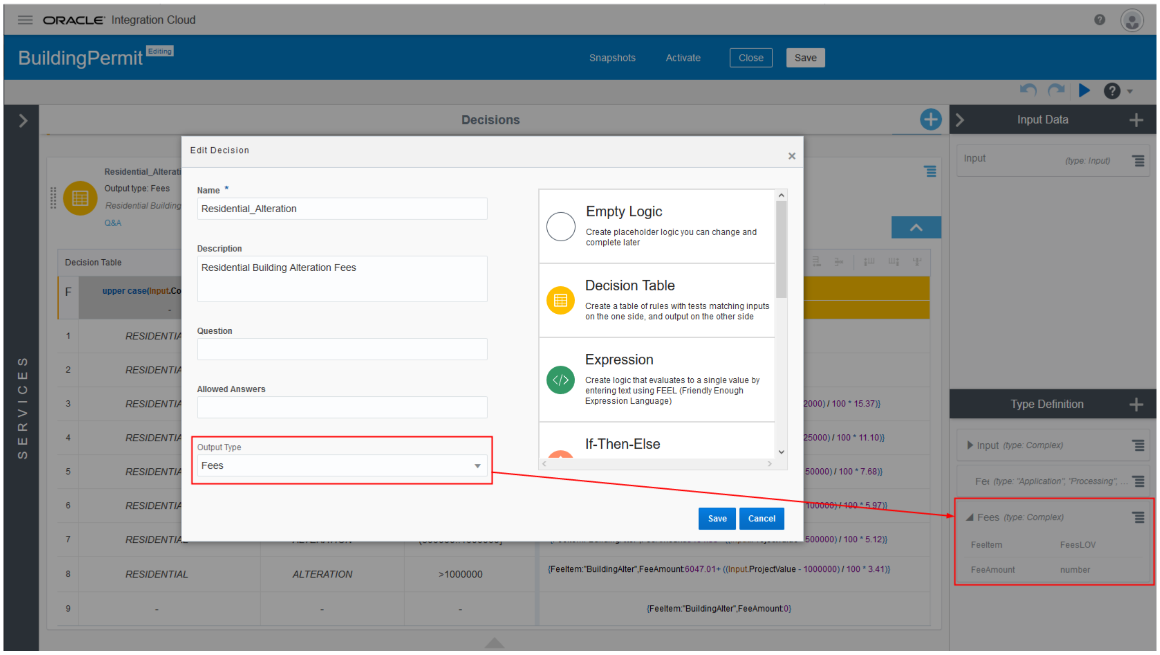
FeeItem	FeesLOV
FeeAmount	number

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.



Edit Decision

Name *
Residential_Alteration

Description
Residential Building Alteration Fees

Question

Allowed Answers

Output Type
Fees

Empty Logic
Create placeholder logic you can change and complete later

Decision Table
Create a table of rules with tests matching inputs on the one side, and output on the other side

Expression
Create logic that evaluates to a single value by entering text using FEEL (Friendly Enough Expression Language)

If-Then-Else

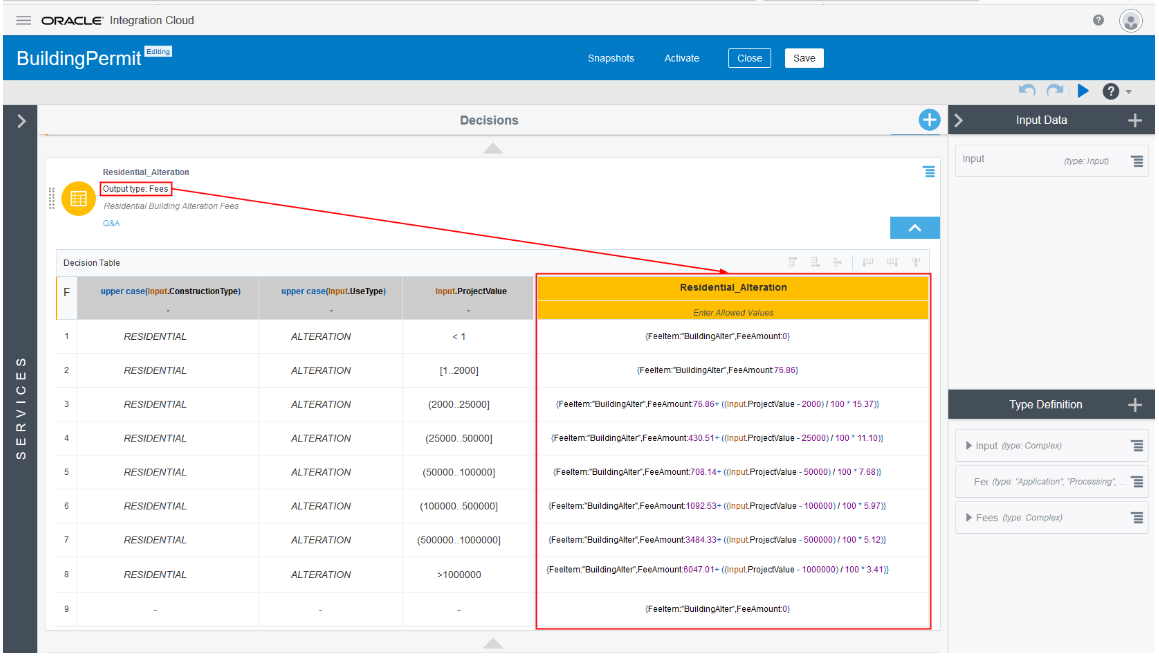
Save Cancel

Fees (type: Complex)

FeeItem	FeesLOV
FeeAmount	number

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.



Residential_Alteration
Output type: Fees
Residential Building Alteration Fees
Q&A

Decision Table

F	upper case(Input.ConstructionType)	upper case(Input.UseType)	Input.ProjectValue	Residential_Alteration
	-	-	-	Enter Allowed Values
1	RESIDENTIAL	ALTERATION	< 1	{FeeItem:"BuildingAlter",FeeAmount:0}
2	RESIDENTIAL	ALTERATION	[1..2000]	{FeeItem:"BuildingAlter",FeeAmount:76.86}
3	RESIDENTIAL	ALTERATION	(2000..25000]	{FeeItem:"BuildingAlter",FeeAmount:76.86+ ((Input.ProjectValue - 2000) / 100 * 15.37)}
4	RESIDENTIAL	ALTERATION	(25000..50000]	{FeeItem:"BuildingAlter",FeeAmount:430.51+ ((Input.ProjectValue - 25000) / 100 * 11.10)}
5	RESIDENTIAL	ALTERATION	(50000..100000]	{FeeItem:"BuildingAlter",FeeAmount:708.14+ ((Input.ProjectValue - 50000) / 100 * 7.68)}
6	RESIDENTIAL	ALTERATION	(100000..500000]	{FeeItem:"BuildingAlter",FeeAmount:1092.53+ ((Input.ProjectValue - 100000) / 100 * 5.97)}
7	RESIDENTIAL	ALTERATION	(500000..1000000]	{FeeItem:"BuildingAlter",FeeAmount:3484.33+ ((Input.ProjectValue - 500000) / 100 * 5.12)}
8	RESIDENTIAL	ALTERATION	>1000000	{FeeItem:"BuildingAlter",FeeAmount:6047.01+ ((Input.ProjectValue - 1000000) / 100 * 3.41)}
9	-	-	-	{FeeItem:"BuildingAlter",FeeAmount:0}

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 the documentation for OIC, *Creating a Service*.

When the Oracle Public Sector Compliance and Regulation service submits a request to the OIC, after running the request against the decision model, the application returns the fee item name and the fee item amount.

Related Topics

- Setting Up Fee Items
- Setting Up Fee Schedules

3 Setting Up GIS

Setting Up Map Profiles

Use map profiles to configure specific instances of map functionality in the system. Profiles set the default extent for the map as well as controlling the availability of certain map options.

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.

Required Map Profiles

Maps appear on various pages in the Oracle system. To configure each map, you need to set up its map profile.

The following tables describe the map profiles that you need to set up and identify the related maps for each profile.

> ⚠ **CAUTION:** Each map is hard-coded to use a map profile with a specific name. So when you set up your map profiles, you must use the exact names given below.

This table lists map profiles for the main maps that appear outside the context of a specific application. There are separate map profiles depending on who accesses this map.

Map Profile	Related Map
PSC_AGENCY_MAIN_MAP	The main map that appears when an agency staff member clicks the **Main Map** icon in the page header.
PSC_GUEST_MAIN_MAP	The main map that appears when an anonymous user clicks the **Explore Your City** tile on the landing page.
PSC_PUBLIC_MAIN_MAP	The main map that appears when a public user who is signed in clicks the **Explore Your City** tile on the landing page. Registered users and anonymous users see different maps because the map for registered users has an additional option for limiting searches to just the user’s own applications. This option is not configured on the map profile, but the existence of two separate maps means that you must set up two separate map profiles.

This table lists additional maps that support applications and inspections.

Map Profile	Related Map
PSC_APO	The property picker map. This map appears in a modal window that is accessed from an application intake form or, for submitted applications, from the application details Property Information page.
PSC_DEFAULT_EXTENT	The map that appears at the top of the application detail pages, the parcel details pages, the address detail pages, and the owner detail pages.

Map Profile	Related Map
PSC_MOBILE_INSPECTION	The map that inspectors see when using mobile inspection functionality.
PSC_PERMIT_LIST	The map that appears on the application list pages when they are in map view. These list pages are the Transactions page for agency staff and the Applications page for registered public users. Both pages have List View and Map View buttons for toggling between views.
PSC_PUBLIC_NOTIFICATION	The map that appears on the Generate Notifications List page. Agency staff uses this page to define a notification area for a hearing. See *Generating a Hearing Notifications List*.

Adding a Map Profile

1. Select **GIS Setup** > **Map Profile.**
2. Click **Add.**
3. Enter the profile name in the **Map Profile** field.

   CAUTION: You must use the exact names that are listed above. Each page that displays a map is hard-coded to look for a specific map profile name.

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

   When you enter the URL, the system asks if you want to use the default map extent from your map service. Typically you will answer Yes.

   The URL itself is not required in the map profile, and saving it to the profile can impact performance. However, for profiles other than PSC_DEFAULT_EXTENT, saving the map URL and turning on the **Enable Layers** switch enables the Identify GIS Information icon on the map toolbar. Users who switch on the Identify GIS Information toggle can click map objects such as parcels to display a pop-up window with object information.

   Tip: Entering the Map Service URL to bring in the default map extent and then clearing the field will improve performance. This is a good option for the PSC_DEFAULT_EXTENT map profile. For other map profiles, you need to weigh the performance impact against the value of giving users the Identify GIS Information option on the maps.

5. Verify or modify the default map extent supplied by the map service URL.

   If it’s necessary to use values other than the default values that come from the map service, your GIS analyst, who understands the map service data, should provide the new values.

   Note: If the PSC_DEFAULT_EXTENT map profile (for maps that appear in the header area of various pages) is able to show a marker related to the map context, it ignores the default map extent from the map profile. For example, in the application detail page for a permit, the map displays a marker for the permit location, if available, rather than showing the default extent from the map profile.

   These fields define the default map extent:

Page Element	Description
X-Min of Default Map View	The top-left X-coordinate of the initial map view extent.
X-Max of Default Map View	The bottom-right X-coordinate of the initial map view extent.
Y-Min of Default Map View	The bottom-left Y-coordinate of the initial map view extent.
Y-Max of Default Map View	The top-right Y-coordinate of the initial map view extent.
Spatial Reference	The geographic coordinate system or map projection used by the mapping service to display the map.

6. Configure the map's user controls.

> **Note:** Most of these settings do not affect the maps that are controlled by the PSC_DEFAULT_EXTENT map profile. The only setting that is relevant to this profile is the **Base Map** field.

Maps can include various widgets that let users manipulate the appearance of the map . These fields let you choose which options are available to users:

Page Element	Description
Enable Default Map View	Indicate whether to display the icon for restoring the map to its initial extent. A user who has zoomed or panned the map clicks this icon to restore the default map area. When this option is enabled, the **Show Default Map View** icon is included on the map tool ribbon.
Base Map	Select the default type of base map. The options are: ○ Dark gray canvas ○ Light gray canvas ○ Imagery with labels ○ National Geographic ○ Topographic ○ Open Street Map ○ Imagery ○ Streets ○ Terrain with labels ○ Oceans
Enable Base Map Gallery	Indicate whether users are allowed to switch to a different base map. When this option is enabled, the **Select Base Map** icon is included on the map tool ribbon.

Page Element	Description
Enable Map Layers	Indicate whether the user is allowed to 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. When this option is enabled, the **Select Layers** icon is included on the map tool ribbon. When this option is enabled and you save a map service URL to the profile, the **Identify GIS Information** icon is included on the map tool ribbon.
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.

7. Click **Save.**

Modifying a Map Profile

1. Select **GIS Setup** > **Map Profile** .
2. On the Map Profiles page, click the row for the profile that you want to modify.
3. To change the default map extent, enter a new map service URL and answer Yes when asked whether to use the associated default extent.

   Alternatively, a GIS analyst can manually update the fields related to the default map extent.
4. As needed, update the settings related to user controls on the related map.
5. Click **Save**.

Deleting Map Profiles

Normally the only reason to delete a map profile is if you accidentally create one with the wrong map name. The correctly-named profiles are required, so you will normally modify the profiles rather than delete them.

To delete a single map profile:

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

To delete multiple map profiles

1. Select **GIS Setup** > **Map Profile.**
2. Click the **Edit** icon.
3. Select the check boxes for the map profiles to delete.
4. Click the **Delete** icon.

Setting Up GIS Attribute Mapping

Use Global Information Systems (GIS) attribute mapping to specify information about your map service parcel layer.

Prerequisites

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

- Publish the map service.
- Ensure that the map service has parcel, address, and owner layers.
- 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. Enter the following parcel layer information on the GIS Attribute Mapping page:

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 as in the 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	Select the parcel layer GIS attribute that provides the unique identifier for each parcel. 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.

3. Enter the following address layer information on the GIS Attribute Mapping page:

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.

Page Element	Description
Parcel Number	Select the address layer GIS attribute that provides the unique identifier for each parcel.

4. Enter the following Owner layer information on the GIS Attribute Mapping page:

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	Select the owner layer GIS attribute that provides the unique identifier for each parcel.

5. Click **Save.**

4 Configuring Oracle Policy Automation

Overview of Oracle Policy Automation Configuration

This topic provides an overview of how Oracle Policy Automation is used within Public Sector Compliance and Regulation and how it is configured.

If your site already has an installation of Oracle Policy Automation, you can integrate its functionality with the permits service. The policy models created in Oracle Policy Automation can act as the logic models running behind questionnaires that citizens 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 Policy Automation 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 Policy Automation 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.
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.

Page Elements	Description

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 Policy Automation.
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 Policy Automation.
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 Policy Automation.
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 Policy Automation. For example: ○ java.lang.String ○ 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 Policy Automation.

Page Elements	Description
Hidden from Policy Modeling	If set to true, then this entry will not be present in the Get MetaData response to Oracle Policy Automation.
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 Policy Automation. 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: ○ String ○ Boolean ○ Decimal ○ Date ○ Date-time ○ 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 Policy Automation.
Enumeration Name	Specifies the ID of the enumeration that defines a field's data type.

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 Policy Automation metadata 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 Policy Automation 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 Policy Automation.

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.

Page Element	Description
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 Policy Automation.

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: ○ String ○ Number ○ Boolean ○ 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.

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 Models Details page, enter the name 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, are read from the metadata model definition.

4. Click **Add** for the Metadata Entity Relationships grid.
5. On the Metadata Entity Relationship Details page, enter these values:

Page Element	Description
Name	Enter a name for the relationship.
Mark as Global Entity	Indicates if this is a global entity for this metadata model.

Page Element	Description
Cardinality with Global Entity	Specify the cardinality relationship that this entity has with the global entity identified for this metadata model, such as one-to-one, one-to-many, and so on.
Policy Modeling Relationship Name	The name of the relationship between two entities as it appears in Oracle Policy Automation.

6. Click **Add** for the Metadata Entity Links grid.
7. On the Metadata Entity Link Details page, enter these values:

Page Element	Description
Source Entity Policy Modeling Name	Enter the policy modeling name for the entity in the source entity model for this link.
Target Entity Model Name	Enter the target entity model for this link.
Target Entity Policy Modeling Name	Enter the policy modeling name for the entity in the target entity model for this link.
Description	Provide any additional information to describe the purpose of this metadata entity link.
Cardinality with Target Entity	Specify the cardinality relationship that this entity has with the target entity model name identified for this metadata link, such as one-to-one, one-to-many, and so on.
Policy Modeling Relationship Name	The name of the relationship between two entities as it appears in Oracle Policy Automation.

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

Related Topics

- Setting Up Metadata Models
- Setting Up Enumerations

Managing Proxy Users

This topic describes how to manage proxy users for enabling integration between Oracle Policy Automation and your Public Sector Compliance and Regulation service.

Oracle Policy Automation connects to your Public Sector Compliance and Regulation service through a provided web service connector named pscOpaWSConnector.

This connector requires proper WS-Security credentials to handle the transactions between the Public Sector Compliance and Regulation service and the Oracle Policy Automation service.

When configuring the connection within the OPA 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 Policy Automation Proxy User (ORA_PSC_OPA_PROXY_USER_DUTY)

This duty role contains the following privilege:

Access Oracle Policy Automation Web Service Connector Privilege (PSC_OPA_WSC_PRIV)

This privilege allows the proxy user to integrate Oracle Policy Automation with your Public Sector Compliance and Regulation service.

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

Related Topics

- Working with Roles in the Security Console
- Managing Roles in Public Sector Compliance and Regulation

Managing OPA Hub

Administrators set up the Oracle Policy Automation required to integrate with Public Sector Compliance and Regulation services.

OPA Setup for Integrating with Public Sector Compliance and Regulation

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 Deployment tile to open the Deployment page. The page lists all the projects currently deployed. Click **Actions** and select the **Authorize Embedded Interviews** button to open the Authorize Hosts page.
3. On the Authorize Hosts page, click the **Add a Host Address** button and enter the Public Sector Compliance and Regulation application host address in the Host address field.
4. 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: http://xmlns.oracle.com/policyautomation/hub/12.2.5/metadata/types
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 startsession 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**.

Page Element	Description
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 OPA, then as part of the Fusion Onboarding process you must have created a user having the following OPA 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 OPA server.

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

Managing OPA Policies for Agency

This topic describes how to set up OPA Policies for your Agency.

1. Navigate to the Agency Information page.
2. Search and select your agency row to open the Agency Information – Details page. Click the Features tab.
3. You will see that your offering is enabled; click the Options link on the row to open the Permit Options page.
4. On the Permit Options page, under the Oracle Policy Automation Definition section enter a value for:

Page Element	Description
Oracle Policy Automation ID	Enter the deployment name listed in the Deployment page. The Deployment page is where the deployment and activation of policy models is managed. **Note:** 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.

5. 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.