

Oracle Financial Services
Analytical Applications
Infrastructure

Process Modelling Framework
(PMF) Orchestration Guide

Release 8.0.6.2.0



DOCUMENT CONTROL

Version Number	Revision Date	Changes Done
1.0	Created Oct 2018	Revamped the PMF User guide content and Orchestration Guide was created.
1.1	Modified Jul 2020	Added the User Role Mapping and Access Rights section to provide information specific to the function roles required to access and use PMF (Doc 31589561).

TABLE OF CONTENTS

1	INTRODUCTION	7
1.1	Key Features of Process Modelling Framework.....	7
1.2	Process Pipeline Flow.....	8
1.1	Prerequisites	8
1.1.1	User Role Mapping and Access Rights	8
1.3	Accessing Process Modelling Framework.....	10
2	PROCESS MODELLER	11
3	COMPONENTS FOR DESIGNING YOUR PROCESS FLOW.....	13
3.1	Transitions	13
3.2	Gateways	13
3.3	Connector	14
3.4	Human Task.....	14
3.5	Service Task	14
3.6	Sub Pipeline.....	14
3.7	Inline Process.....	14
3.8	OFSAA Widgets	14
4	ARTIFACTS OF PROCESS MODELLING.....	15
4.1	Application Package.....	15
4.2	Data Fields.....	15
4.2.1	Adding a Data Field.....	15
4.2.2	System Data Fields	17
4.2.3	AOM Data Field	18
4.3	Application Rules	19
4.3.1	SQL Application Rule:	19
4.3.2	Stored Procedure Application Rule:.....	20
4.3.3	For Function Application Rule:	21
4.3.4	For Java Application Rule:	23
4.3.5	For Java External API Application Rule:.....	24
4.3.6	For Outcome Rules Application Rule.....	25
4.3.7	Expression Application Rule.....	25
4.3.8	Rest Service Application Rule	26
4.3.9	For Attribute Expression Application Rule	28
4.3.10	For Advanced Attribute Expression Application Rule.....	29

4.3.11	For JSON Path Expression Application Rule	30
4.3.12	JSON Read From DB Application Rule	32
4.3.13	JSON Write To DB Application Rule.....	33
4.4	Configuring Application Object Model (AOM).....	34
4.4.1	Attribute Types.....	35
4.4.2	AAI_AOM_APP_COMP_ATTR_TL Table	35
4.4.3	Usage of AOM Attributes in Run Pipeline	36
5	DESIGNING A PIPELINE	37
5.1	Orchestration of a Business Pipeline	37
5.1.1	An Example of a Business Pipeline	37
5.1.2	Creating a Business Pipeline.....	38
5.2	Run Pipeline.....	39
5.2.1	An Example of Run Pipeline.....	39
5.2.2	Creating a Run Pipeline	40
5.2.3	Design your Run Pipeline using OFSAA Widgets	41
5.2.4	Design your Run Pipeline using Sub Pipeline.....	41
5.2.5	Design your Run Pipeline using Inline Process	41
5.2.6	Design your Run Pipeline using combinations of OFSAA Widgets, Sub Pipeline and Inline Process	41
5.2.7	Executing Run Pipeline	41
6	HUMAN TASKS	43
6.1	How to Use Human Task	43
6.1.1	Activity Tab	44
6.1.2	Action Tab for Creating Tasks/Notification.....	44
6.2	Additional Functionalities for Human Tasks	46
6.2.1	Setting Task Expiry	46
6.2.2	Setting Task Escalation.....	47
6.2.3	Setting Reminder for Your Task	48
7	SERVICE TASKS	49
7.1	How to Use a Service Task.....	49
7.1.1	Activity Tab	49
7.1.2	Implementation Tab	49
8	CONFIGURING OFSAA TASKS IN YOUR PROCESS FLOW	52
8.1	How to Use an OFSAA Widget	52
8.1.1	Dynamic Parameters for OFSAA Widgets	53
9	ORCHESTRATING EXTERNAL MODELS/COMPONENTS IN YOUR PROCESS FLOW	59

9.1	How to invoke External Model through Web Service	59
9.1.1	Data Preparation.....	59
9.1.2	Webservice Invocation	60
9.1.3	Data Extraction	62
10	CONFIGURING CUSTOM COMPONENTS	64
11	EXECUTING PARALLEL TASKS	65
11.1	How to Use Parallel Gateways.....	66
12	CALLING ANOTHER PIPELINE FROM YOUR PARENT PIPELINE.....	69
12.1	How to Configure Sub Pipeline	69
13	INLINE PROCESS	71
13.1	An Example of Inline Process	71
13.2	How to Add an Inline Process	72
14	CONFIGURING EMAIL FOR HUMAN TASKS	74
14.1	AAI_EMAIL_CONFIG Table.....	74
14.2	AAI_WF_APP_PACKAGE_B Table.....	74
14.3	AAI_WF_APP_REGISTRATION Table.....	75
14.4	AAI_WF_ACTIVITY_TASK_B Table.....	75
14.5	AAI_USER_PREFERENCE Table.....	75
14.6	AAI_WF_EMAIL_TEMPLATE Table	76
14.7	AAI_WF_BULK_MAIL_TRIGGER Table.....	76
14.8	CSSMS_USR_PROFILE Table.....	76
14.9	AAI_MAIL_AUDIT_TRAIL Table	76
15	PROCESS MONITOR	77
15.1	Monitoring a Business Process	77
15.2	Viewing Activity Logs	79
16	INVOKING PMF PIPELINE	81
16.1	Application UI.....	81
16.2	Within PMF Summary Screen UI	82
16.2.1	Using Test Workflow	82
16.2.2	Using Execute Run	82
16.3	Operations Module.....	82
16.4	Command Line Execution	83
17	APPENDIX A.....	84

- 17.1 Enabling Proxy for REST Service Application Rule..... 84
- 17.2 Configuring Group Approval for Human Tasks..... 84
 - 17.2.1 Configuring Parallel Group Approval 84
 - 17.2.2 Configuring Sequential Group Approval 86
- 17.3 JsonPath Expressions..... 87
 - 17.3.1 Operators 88
 - 17.3.2 Functions 88
 - 17.3.3 Filter Operators 88
 - 17.3.4 Path Examples 89
- 17.4 Delegation..... 90
 - 17.4.1 Adding a Delegate..... 91
 - 17.4.2 Viewing Delegation 93
 - 17.4.3 Modifying Delegate Details..... 93
 - 17.4.4 Revoking Delegation 94
 - 17.4.5 Deleting Delegation..... 94

1 Introduction

Process Modeling Framework (PMF) is a design and execution framework that enables Process Pipeline developers to implement various Pipelines modelled by business analysts. Process Pipeline developers use the framework to orchestrate the Business Pipelines and Run Pipelines within OFSAA, and also to design the artifacts that participate in the Pipelines, in order to complete their implementation.

The Process Modeling Framework consists of Process Modeling components for modeling Pipelines and Process Monitor components for monitoring instantiated Pipelines of OFSAA applications.

See [Process Flow](#) for more information on how these tools fit into the Pipeline design and implementation.

Process Modeller aids in representing the various artifacts required for modeling and provides implementation details of the OFSAA process artifacts.

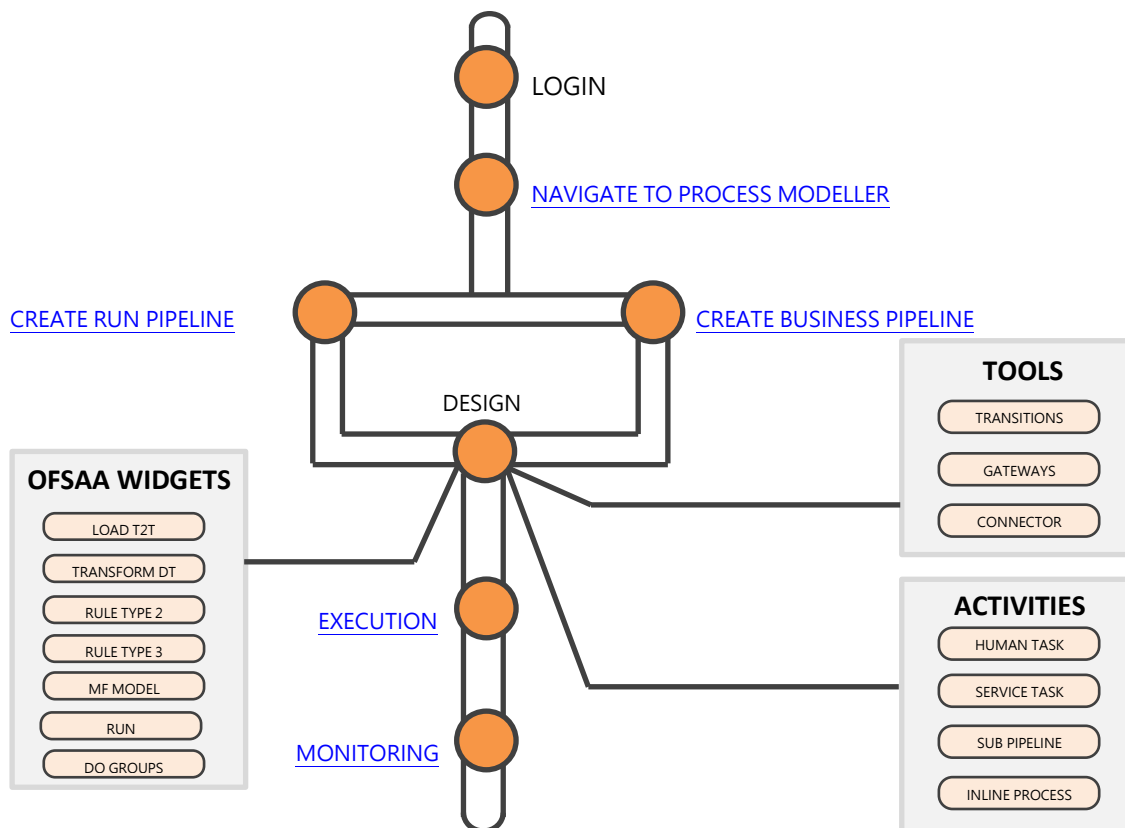
- OFSAA Process Pipeline or Run Pipeline
- Reusable process components like Inline Process / Sub Pipeline
- Process data (Data Fields)
- Implementation of various types of Human Tasks / Service Tasks
- Business Rules (Application Rules)
- Various External services implementations and other artifacts needed for complex implementations
- Configuring Notifications

1.1 Key Features of Process Modelling Framework

- Support for visual modeling of the pipelines.
- Support for registration of Process /Activity/ Transition Logic implementation, separated from the modeling itself.
- Built-in orchestration engine (included within OFSAA's runtime) for task execution (interactive model as opposed to the batch model supported through Rule Run Framework).
- Published interface for abstraction of task implementation.
- Representation of the pipeline-routing rule logic in Java/ PL-SQL / Web-service.
- Stitching of OFSAA Components within the Process Pipeline
- Orchestration and execution of RRF Run
- Reminder, Escalation, and Expiry of tasks

- Registration of Custom Widgets
- Process Monitoring Admin Tool to view the execution Process Instances.

1.2 Process Pipeline Flow



1.1 Prerequisites

The following is the prerequisite to access and perform functions in the PMF user-interface:

- [User Role Mapping and Access Rights](#)

1.1.1 User Role Mapping and Access Rights


User access to the PMF UI and ability to perform functions in it is dependent on the mapping of the user profile to the roles in the OFS AAI application and the access rights assigned.

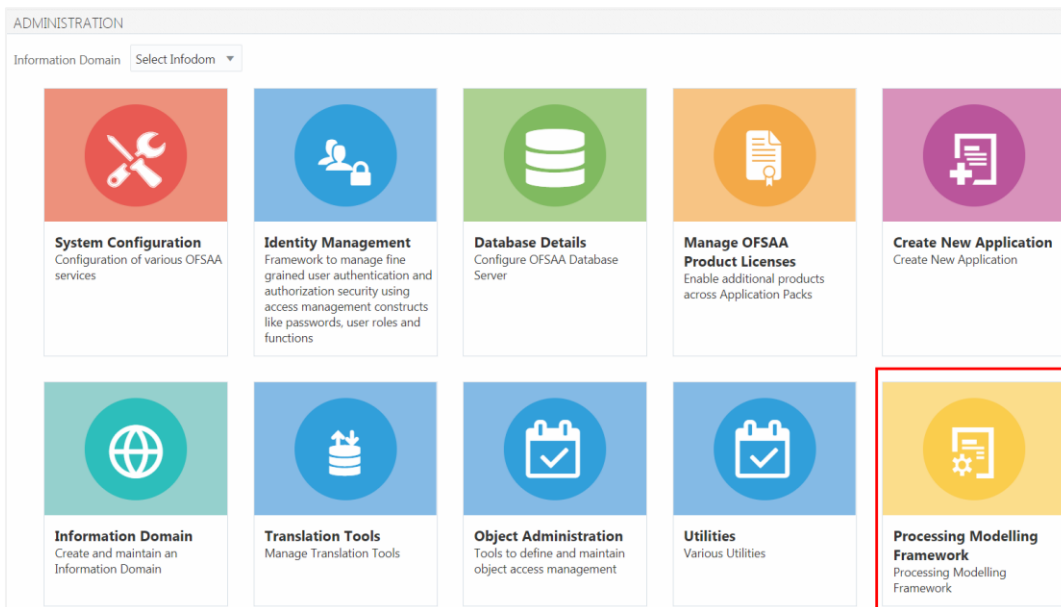
The following user role mapping is mandatory for PMF:

Role Code	Role Name	Functionality
WFACC	Workflow Access	Assign this role to the user to access the Process Modeller menu from the Navigation Tree. Note: The mapping of this role does not allow view, edit, and add actions.
WFMAcc	Workflow Monitor Access	Assign this role to the user to access the Process Monitor window. Note: The mapping of this role does not allow view, edit, and add actions.
WFREAD	Workflow Read	Assign this role to the user to view the PMF workflow.
WFWRITE	Workflow Write	Assign this role to the user to perform view, edit, and add actions in PMF.

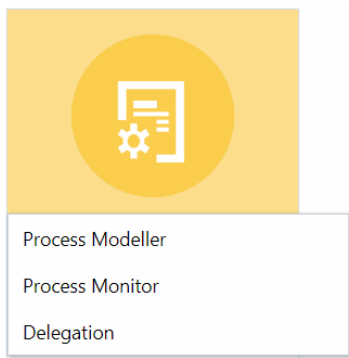
NOTE: For administrators, ensure that they are mapped to all the roles described in the preceding table to allow them to perform all types of operations in PMF.

1.3 Accessing Process Modelling Framework

From the OFSAA Landing screen, click  Administration.

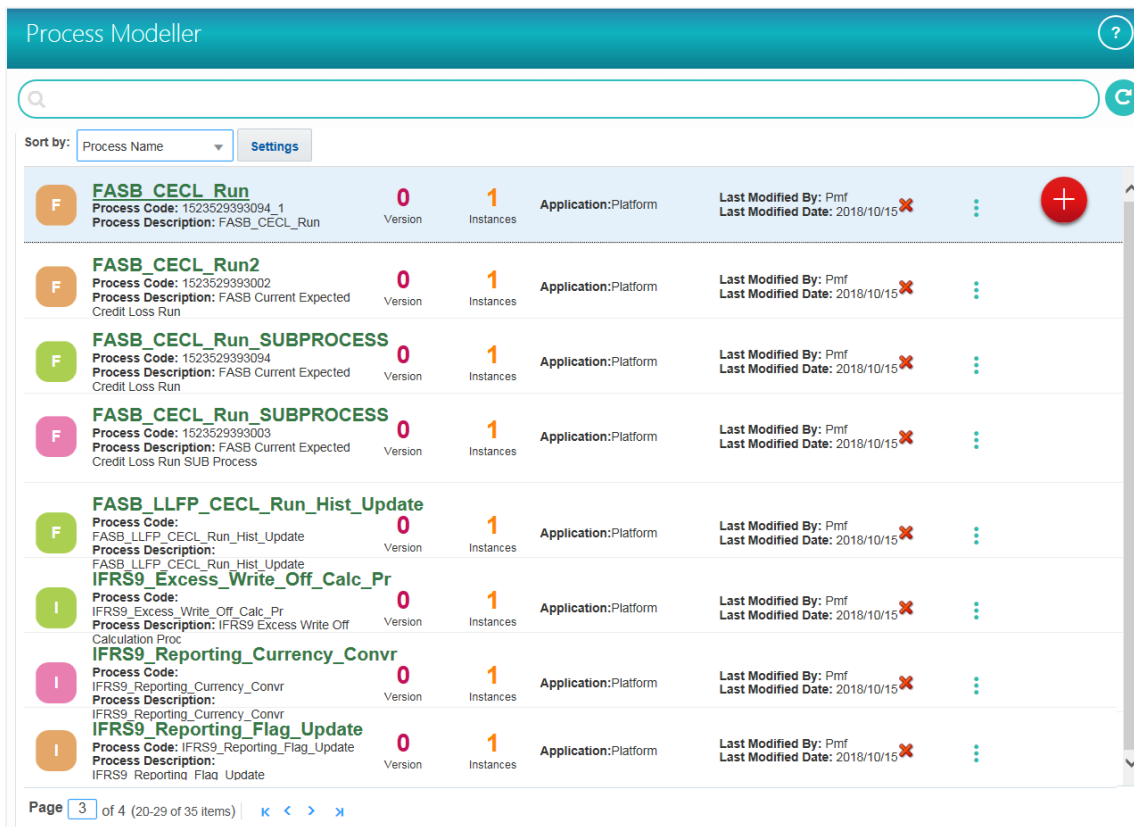


Click **Process Modelling Framework** tile to display the sub menu.






- Click **Process Modeller** to launch the Process Modeller.
- Click **Process Monitor** to monitor currently running processes.
- Click **Delegation** to launch the Delegation framework.

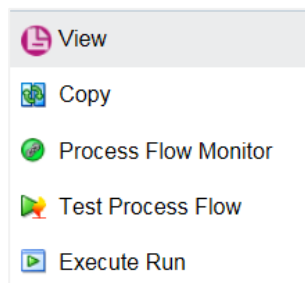
2 Process Modeller




The *Process Modeller* window displays the existing Business Process Pipelines and Run Pipelines with the details such as Process ID, Process Name, Process Description, Version, Instance, Application, and Last Modified details.

You can do the following tasks from this window:

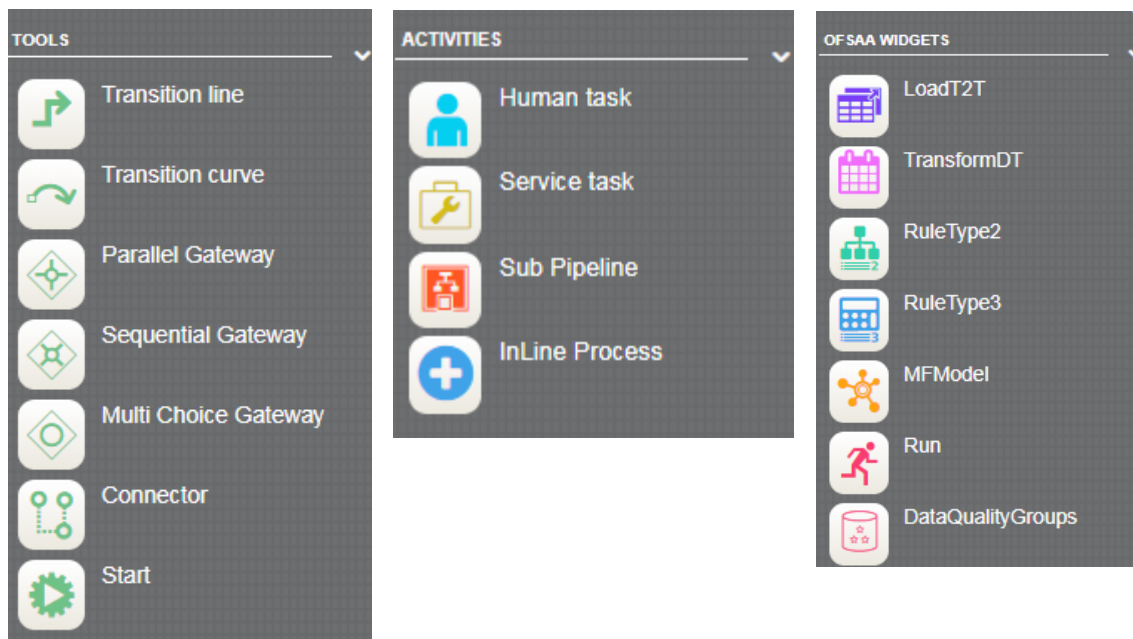
- Click  to create a new Pipeline.
- Click the Process Name link to launch and edit the Process Flow.
- Click  to delete a Pipeline.
- Click  to view the following sub menu:





- Click **View** to see the process flow.
- Click **Copy** to create a new Pipeline with the same process flow.
- Click **Process Flow Monitor** to monitor the Pipeline.
- Click **Test Process Flow** to test/check whether the Business Pipeline you designed works as expected.
- Click **Execute Run** to execute a Run Pipeline.
- Using the *Search* grid, you can search for a specific Pipeline based on Process ID, Process Name, Application or Version. Click  to reset the Search fields.
- You can sort the Pipelines based on Process ID, Process Name, or Application. Click the **Sort by** drop-down and select the attribute by which you want to sort.

3 Components for Designing Your Process Flow

The Process Flow tab has a floating toolbar and a drawing canvas. Drawing canvas is used to design the Process flow with the Tools, Activities, and OFSAA Widgets available in the floating toolbar.



3.1 Transitions

Transition is used to control the flow between various components in the Process flow. Transition connects two activities and the flow is configured based on conditional expression or decision rule. You can use Transition Line  if you want to use straight line to connect the activities or Transition Curve  if you want to use curved lines, based on your requirement.

3.2 Gateways

Split refers to a condition where an incoming transition is split to multiple transitions. In Merge, multiple incoming transitions are merged to a single transition. The Splitting and Merging of Activities are modeled through Gateways. Gateway can be Sequential, Parallel, or Multi Choice.

- **Parallel Gateway:** A Parallel gateway is used when you want to have multiple transitions/flows which should be executed in parallel.
- **Sequential Gateway:** A Sequential gateway is used when you want to have multiple transitions/flows which should be run in sequential.
- **Multi Choice Gateway:** A Multi Choice gateway is used when you want to execute multiple transitions/flows based on decision rule.

3.3 Connector

Connector helps to connect two activities with different path (other than the default), in case if the default path is overlapping with some existing flow.

3.4 Human Task

Human task requires human intervention to move to next Activity. For more information on stitching human tasks in your Process flow, see [Human Tasks](#) section.

3.5 Service Task

Service task typically invokes an application component (for example, activity to invoke a business rule to calculate certain threshold). For more information on stitching service tasks in your Process flow, see [Service Tasks](#) section.

3.6 Sub Pipeline

Sub Pipeline provides reusability of Pipelines. Using Sub Pipeline component, you can call another Pipeline from your parent Pipeline. For more information on how to use Sub Pipeline, see [Calling another Pipeline from Your Parent Pipeline](#) section.

3.7 Inline Process

Inline Process is used to group a set of processes within a process flow. It is a process flow within another process flow, which is not reusable like Sub Pipeline. For more information, see [Inline Process](#) section.

3.8 OFSAA Widgets

OFSAA widgets are used to execute OFSAA components like T2T definitions, PLC definitions (DT), Rules (Classification Rule and Computation Rule), Models in EMF, RRF Runs, Data Quality Groups, and RRF Processes through Process Modeller. For more information, see [Configuring OFSAA Tasks in Your Process Flow](#) section.

You can register a new component by entering details in the AAI_WF_COMPONENT_REGISTRATION table. For more information, see [Configuring Custom Components](#) section.

4 Artifacts of Process Modelling

Before you start designing your Pipeline, it is required to get introduced to some artifacts of Process Modelling Framework.

4.1 Application Package

Application package is a concept used to group Pipelines, Application Rules, and Data Fields which are required for an Application. When you create a Pipeline, you should select the Application Package in which the Pipeline needs to be available. Similarly, when you define a Data Field or an Application Rule, you can set it to be available across Pipelines created in that Application Package.

You can add a new package by adding a new entry in the AAI_WF_PACKAGE_B table.

V_APP_PACKAGE_ID	V_APP_PACKAGE_DESC	V_LANGIN	V_DEFINITION_PAGE_URL	V_IS_EMAIL_REQUIREI
0	Global			N
10	Business Restructure		Restructure/manage_grid.jsp?userId={ASSIGNEEUSERS}&locz	Y
100	Platform			Y
11	Questionnaire		solution/abc_qtnr/QtnrRedirectFrmPMFInbox.jsp?appCode={ap	Y

V_APP_PACKAGE_ID – Enter a unique application package ID.

V_APP_PACKAGE_DESC – Enter a description for the application package.


V_DEFINITION_PAGE_URL – Enter the URL of the definition page of the Application.

V_IS_EMAIL_REQUIRED – Set this as Y for configuring email at Application Level.

4.2 Data Fields

Data Field, which is also known as Process Variable, helps Process Pipelines to access and store information from outside application. Often the process flow is based on the value of this information. In other cases, this information is the result of running the tasks in the Pipeline.

4.2.1 Adding a Data Field

1. From the *Process Modeller* window, click  corresponding to the Pipeline for which you want to add a Data Field. The *Process Flow* tab is displayed.
2. Select the *DataFields* tab.

Select	Name	Description	Type	Default Value	Is Mandatory
<input type="radio"/>	BUCKETCONVENTION	Bucket Conversion	AOM		Y
<input type="radio"/>	CONSOTYPE	Consolidation Type	AOM		Y
<input type="radio"/>	test	Datafield	STRING		
<input type="radio"/>	WF_ENTITYID	ENTITYID	STRING		
<input type="radio"/>	FIC_MIS_DATE	FIC MIS Date	AOM		Y
<input type="radio"/>	WF_INFODOM_CODE	INFODOM_CODE	STRING		
<input type="radio"/>	WF_INSTANCE	INSTANCE	STRING		
<input type="radio"/>	INTERPOLMETHOD	Interpolation Method	AOM		Y
<input type="radio"/>	WF_LOCALE	LOCALE	STRING		
<input type="radio"/>	LE	Legal Entity	AOM		Y
<input type="radio"/>	N_RUNTYPE_SK	N_RUNTYPE_SK	INT	3	
<input type="radio"/>	WF_OBJECT_ID	OBJECT_ID	STRING		
<input type="radio"/>	WF_OBJECT_NAME	OBJECT_NAME	STRING		
<input type="radio"/>	WF_OBJECT_TYPE	OBJECT_TYPE	STRING		
<input type="radio"/>	WF_OUTCOME_ID	OUTCOME_ID	STRING		
<input type="radio"/>	PD_INTERPOLATION_MTHD	PD Interpolation Method	AOM		Y
<input type="radio"/>	WF_PROCESS_ID	PROCESS_ID	STRING		
<input type="radio"/>	RCY	Reporting Currency	AOM		Y
<input type="radio"/>	V_RUN_MAIN_DESC	Run Execution Description	AOM		Y
<input type="radio"/>	WF_SEGMENT_CODE	SEGMENT_CODE	STRING		

3. Click **Add**. The *Data Field Details* window is displayed.

Data Field Details ✕

Datafield Code ?

Datafield Description ?

Datafield Type ? ▼

Initial Value ?

Is Mandatory ? ▼

Scope ? ▼

4. Enter the details as given in the table:

Field Name	Description
Data field Code	Enter the Variable Name/Code, which needs to be used by application to read or write in to this variable. This field is non translatable.
Data field Description	Enter a brief description of the Data field.
Data field Type	Enter the Data Field type. The supported types are String, Integer, Int, Float, AOM (Application Object Model), and Boolean. For more information on where we use AOM Data Field type, see AOM Data Field section.
Initial Value	Enter the default value for the Data.
Is Mandatory	This field is applicable for Data Field Type of AOM. By default, for AOM Data Field, this is selected as Yes. Select No if you do not want this parameter to be displayed as Execution Parameter for the Run Pipeline.
Scope	Select the scope of the Data Field from the drop-down list. The options are: <ul style="list-style-type: none"> ▪ Process- Select Process if you want to use the Data Field only in the current process. ▪ Package- Select Package if you want to use the Data Field across all the processes in the package.

5. Click **OK**.

4.2.2 System Data Fields

Some data are tracked internally by the System using a predefined set of Data Fields such as Status of Process. You can access these activity instance attributes in the same way you access regular data objects, but you cannot assign them new values.

System Data Field	Description
WF_RUNSK	Holds the Runsk value.
WF_MISDATE	Holds the MIS date.
WF_INFODOM_CODE	Holds the Information domain code.
WF_USERID	Holds the UserId who invoked the Workflow.
WF_LOCALE	Holds the locale.

System Data Field	Description
WF_OBJECT_NAME	Holds the Object name.
WF_INSTANCE	Holds the Process Instance Id.
WF_SEGMENT_CODE	Holds the Segment Code.
WF_PROCESS_ID	Holds the Process Id.
WF_OUTCOME_ID	Holds the Outcome Id.
WF_OBJECT_ID	Holds the Object Id.
WF_ENTITY_ID	Holds the Entity Id.
WF_OBJECT_TYPE	Holds the Object Type.
WF_TASK_STATUS	Holds the Task status.
WF_TASK_RESPONSE	Holds the Task response.
WF_CALLBACK_TASK_STATUS	Holds the callback Task status.
WF_STATUS	Holds the current state of the Workflow. For example, Draft, PendingApproval and so on.

4.2.3 AOM Data Field

The AOM Data Fields are automatically created from the entries in the `AAI_AOM_APP_COMP_ATTR_MAPPING` table. These Data Fields, which are marked as mandatory, will be displayed in the *Select Run Parameters* window while [executing Run Pipeline](#). For configuring AOM Data Fields, see [Configuring Application Object Model \(AOM\)](#) section.

4.3 Application Rules

The Application or API Rule is the interface between the process engine and the application, including any parameters to be passed.

Based on their usage these are categorized into three types.

- **Execution Rule:** These are Business Logic executed as Task by an Activity.
- **Decision Rule:** This rule returns Boolean value “True/False”, used in decision making during split/branching of transition.
- **Selection Rule:** This rule fetches some value, useful to get value dynamically from a table or other source.

For example, `select v_created_by from fct_expenses where id=101`

4.3.1 SQL Application Rule:

This Application Rule is used to execute any SQL queries in the Process Flow.

The screenshot shows a 'Rule Details' configuration window with the following fields:

- Name:** Process_Name_Data
- Rule Type:** Execution Rule
- Execution Type:** SQL
- Implementation Detail:** SELECT PROCESSNAME FROM ofsaatm.Report_Links WHERE REPORT_TYPE='MODEL_RUN'
- Return Parameter:** PROCESS_NAME
- Scope:** Package

A 'Save' button is located at the bottom of the form.

- Enter a unique **Name** for the Application Rule.
- Select the **Rule Type** from the drop-down list. The SQL Application Rule can be used as Decision Rule, Execution Rule or Selection Rule based on your requirement.
- Enter the Implementation Details.
 - **Decision Rule-** For Decision Rule the SQL Statement should return 'PASS' for success condition.

For example, `select 'PASS' from dual where {EXPENSES} <= {THRESHOLD}`

Note: {EXPENSES}, {THRESHOLD} are user defined Data Fields.

- **Execution Rule-** For Execution Rule the SQL Statement can be any DML statement.
For example, `update fct_expenses set expenses={EXPENSES} where id = {WF_ENTITYID}`
- **Selection Rule-** For Selection Rule the SQL Statement should be a Select statement which returns a list of values.
For example, `select v_created_by from fct_expenses id = {WF_ENTITYID}`
- Select the Data Field which will receive the **Return Parameter** of the SQL Rule, from the drop-down list.
 - For ExecutionRule type, the business logic is implemented in the method and the parameter value returned from the SQL Rule is saved in the mapped Data Field.
 - In case of SelectionRule type, the Java method should be a String value.
 - In case of DecisionRule type, the Java method should return Boolean values "True/False".
- Select the **Scope** as **Process** to use the Application Rule only in the current process or **Package** to use the Application Rule across all the processes in the package.

4.3.2 Stored Procedure Application Rule:

This Application Rule is used to call a Stored Procedure in your Process Flow.

The screenshot shows a 'Rule Details' form with the following fields and values:

- Name:** DQ EXE RULE
- Rule Type:** Execution Rule
- Execution Type:** Stored Procedure
- Procedure Name:** dqcheck
- InputParameters:** DQ STATUS
- Parameter Mode:** ["OUT"]
- Scope:** Package

A 'Save' button is located at the bottom of the form.

- Enter a unique **Name** for the Application Rule.

- Select the **Rule Type** from the drop-down list. The Stored Procedure Application Rule can be used as Decision Rule, Execution Rule or Selection Rule based on your requirement.
- Enter the Stored **Procedure Name** which you want to call in your process flow.
- You can pass **Input Parameters** for the stored Procedure using Data Fields. Select the required Data Fields from the drop-down list.
- Enter the **Parameter Mode** in JSON format. For example, suppose you have given 3 parameters as input parameters, enter parameter mode as ["IN","IN","OUT"]. During Execution of Stored Procedure,
 - In case of Decision Rule type, the first return parameter should return value 'PASS' for success evaluation.
 - In case of Selection Rule type, the first return parameter value is taken as Selection data.
 - In case of Execution Rule, the procedure return OUT parameter value overwrites the current value of respective mapped Data Field.
- Select the **Scope** as **Process** to use the Application Rule only in the current process or **Package** to use the Application Rule across all the processes in the package.

4.3.3 For Function Application Rule:

This Application Rule is used to call Database functions in your Process Flow.

The screenshot shows a 'Rule Details' dialog box with the following fields and values:

- Name**: Getting Additional Param
- Rule Type**: Execution Rule
- Execution Type**: Function
- Function Name**: FN_DQ_ADDITIONAL_PARAMS
- InputParameters**: OBJECT_ID, USERID, OBJECT_NAME
- Parameter Mode**: (empty)
- Return Parameter**: TASK_STATUS
- Scope**: Process

An 'Ok' button is located at the bottom center of the dialog.

Field Name	Description
Name	Enter a unique name for the Application Rule.
Rule Type	Select the rule type from the drop-down list. The available rule types are Decision Rule, Execution Rule and Selection Rule.
Execution Type	Displays the Application Execution Type as Function.
Function Name	Enter the Function Name.
Input Parameters	Select the list of Data Fields which will be passed as input parameters, from the drop-down list.
Parameter Mode	<p>Enter the Parameter Mode in JSON format. For example, suppose you have given 3 parameters as input parameters, enter parameter mode as ["IN","IN","OUT"].</p> <p>During Execution of Function,</p> <ul style="list-style-type: none"> ▪ In case of Decision Rule type, the first return parameter should return value 'PASS' for success evaluation. ▪ In case of Selection Rule type, the first return parameter value is taken as Selection data. ▪ In case of Execution Rule, the procedure return OUT parameter value overwrites the current value of respective mapped Data Field.
Return Parameter	<p>Select the Data Field which will receive the return parameter of the Java function, from the drop-down list.</p> <p>For ExecutionRule type, the business logic is implemented in the method and the parameter value returned from the Java method is saved in the mapped Data Field.</p> <p>In case of SelectionRule type, the Java method should be a String value.</p> <p>In case of DecisionRule type, the Java method should return Boolean values "True/False".</p>
Scope	<p>Select the scope of the Application Rule from the drop-down list. The options are:</p> <ul style="list-style-type: none"> ▪ Process- Select Process if you want to use the Application Rule only in the current process. ▪ Package- Select Package if you want to use the Application Rule across all the processes in the Application package.

4.3.4 For Java Application Rule:

This Application Rule is used to call Java functions in your Process flow.

The screenshot shows a 'Rule Details' dialog box with the following fields and values:

- Name**: CallECMEventCreationService
- Rule Type**: Execution Rule
- Execution Type**: JAVA
- Implementation Detail**: com.ofss.fccm.fraud.main.ResponseMain
- Return Parameter**: ENTITYID
- Scope**: Process

A 'Save' button is located at the bottom center of the dialog.

- Enter a unique **Name** for the Application Rule.
- Java Rule supports only Execution Rule Type.
- Enter the complete java class name which implements the Interface :

```
com.ofs.aai.service.wf.external.base.Activity
```

The implementation class has to override the method with the Business Logic.

```
executeTask(List<Data Field> Data Fields)
```

The Data Fields are passed by reference, so changes can be made in Data Fields value directly by the implementation class, which will be recognized by the WorkflowEngine.

- Select the Data Field which will receive the **Return Parameter** of the Java Rule, from the drop-down list.
 - For ExecutionRule type, the business logic is implemented in the method and the parameter value returned from the SQL Rule is saved in the mapped Data Field.
 - In case of SelectionRule type, the Java method should be a String value.
 - In case of DecisionRule type, the Java method should return Boolean values "True/False".
- Select the **Scope** as **Process** to use the Application Rule only in the current process or **Package** to use the Application Rule across all the processes in the package.

NOTE: The class and its dependent file (or jar) needs to be available in webcontainer class path.
Eg. <TOMCAT_HOME>/webapps/<context>/WEB-INF/lib/ <forecast.jar>

4.3.5 For Java External API Application Rule:

This Application Rule is used to call Java External API in your process flow. You need to specify the Class Name and the method of the API.

The screenshot shows a 'Rule Details' form with the following fields:

- Name:** ForecastPD
- Rule Type:** Execution Rule
- Execution Type:** Java External API
- Class Name:** com.Forecast
- Method:** forecastPD
- InputParameters:** INFODOM_CODE
- Return Parameter:** STATUS
- Scope:** Process

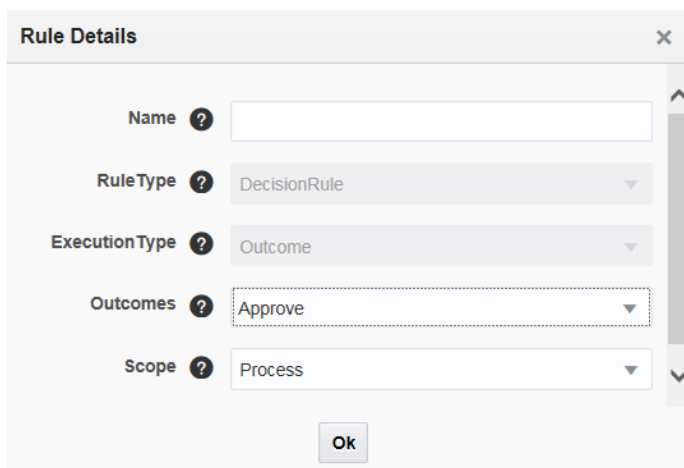
A 'Save' button is located at the bottom of the form.

- Enter a unique **Name** for the Application Rule.
- Select the **Rule Type** from the drop-down list. The Java External API Application Rule can be used as Decision Rule, Execution Rule or Selection Rule based on your requirement.
- Enter the complete java **Class Name** which implements the Business logic.
- Enter the **Method** which you want to execute.
- You can pass **Input Parameters** for the method using Data Fields. Select the required Data Fields from the drop-down list.
- Select the Data Field which will store the **Return Parameter** of the method, from the drop-down list.
- Select the **Scope** as **Process** to use the Application Rule only in the current process or **Package** to use the Application Rule across all the processes in the package.

NOTE: The class and its dependent file (or jar) needs to be available in webcontainer class path.

For example, <TOMCAT_HOME>/webapps/<context>/WEB-INF/lib/
<forecast.jar>

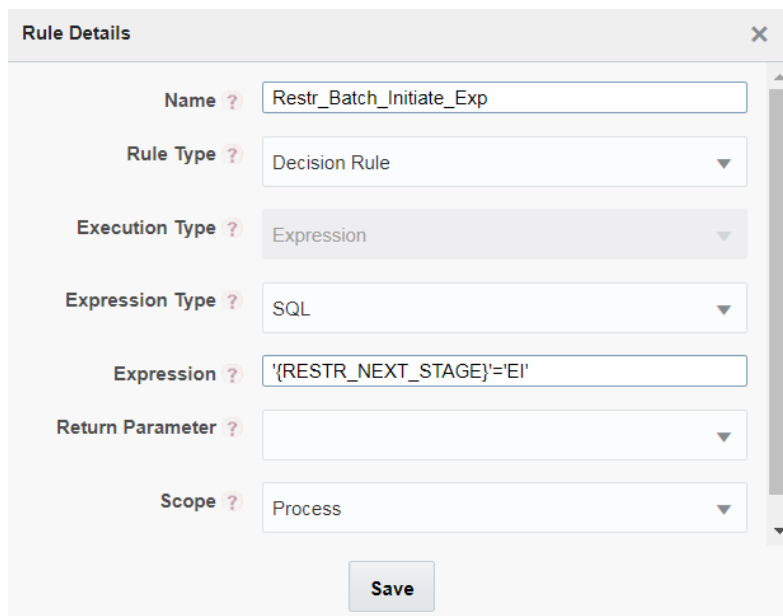
4.3.6 For Outcome Rules Application Rule



- Enter a unique **Name** for the Application Rule.
- Select the outcome for which you want to add the rule.
- Select the **Scope** as **Process** to use the Application Rule only in the current process or **Package** to use the Application Rule across all the processes in the package.

4.3.7 Expression Application Rule

This is same as SQL execution type. You need to specify only the where clause in the Expression field. It can be any SQL expressions including 'AND'/'OR'.



- Enter a unique **Name** for the Application Rule.
- Select the **Rule Type** from the drop-down list. The Expression Application Rule can be used as Decision Rule, Execution Rule or Selection Rule based on your requirement.
- Select Expression Type as SQL to use SQL expressions or JSON to use JSON expressions.
- Enter the **Expression** in SQL format or JSON format.
- Select the Data Field which will receive the **Return Parameter** of the Expression, from the drop-down list.
 - For ExecutionRule type, the business logic is implemented in the method and the parameter value returned from the Expression is saved in the mapped Data Field.
 - In case of SelectionRule type, the expression should be a String value.
 - In case of DecisionRule type, the expression should return Boolean values “True/False”.
- Select the **Scope** as **Process** to use the Application Rule only in the current process or **Package** to use the Application Rule across all the processes in the package.

4.3.8 Rest Service Application Rule

This Application Rule is used to call external components in your Process flow.

The screenshot shows a 'Rule Details' form with the following fields and values:

- Name**: ModelWebService
- RuleType**: ExecutionRule
- ExecutionType**: Rest Service
- Method Type**: POST
- Url**: http://10.184.154.193:8001/ofsaai/rest-api/v1/ModelWebService/ForecastModelPost
- Query Param**: (empty)
- Headers**: (empty)
- Data**: {"owner": {"id": 3730}, "scenario_name": "Baseline", "forecast_periods": 2, "data": ~~JSON_DB_DATA~~}
- Return Parameter**: TASK_RESPONSE
- Scope**: Process
- Is Proxy Required**: Yes

A 'Save' button is located at the bottom of the form.

- Enter a unique **Name** for the Application Rule.

- Select the **Rule Type** from the drop-down list. The Rest Service Application Rule can be used as Decision Rule, Execution Rule or Selection Rule based on your requirement.
- Select the **Method Type** as GET or POST from the drop-down list.
- Enter the REST **URL** that needs to be called.

For example, <IP Address/hostname of the Web Server >:<servlet port>/<context name>/restPMF/PMFService/startWorkflowProcess (A rest URL to start the workflow)

- Enter the **Query Parameters** that needs to be passed to the rest API.

For example, <http://example.com/foo?bar>

- Enter any **Headers** that needs to be passed to the rest API.

For example, "content-type": "application/json"

- Enter if any actual **Data** that needs to be passed to the rest API. Data can be of type RAW, JSON, Form Data and so on.

For example, "{\n \"objectid\": \"912\", \n \"objecttype\": \"1000\", \n \"infodom\": \"OFSCAPADQINFO\", \n \"segment\": \"OFSCAPADQINFO\", \n \"userid\": \"ORMUSER\", \n \"locale\": \"en_US\", \n \"securityMap\": {}, \n \"applicationParams\": {\n \"testparam\": \"value1\", \n \"testparam2\": \"value2\" \n } \n \n }

- Select the Data Field which will receive the **Return Parameter** of the Rest Service Rule, from the drop-down list.
 - For ExecutionRule type, the business logic is implemented in the method and the parameter value returned from the Application Rule is saved in the mapped Data Field.
 - In case of SelectionRule type, the Application Rule should be a String value.
 - In case of DecisionRule type, the Application Rule should return Boolean values "True/False".
- Select the **Scope** as **Process** to use the Application Rule only in the current process or **Package** to use the Application Rule across all the processes in the package.
- Select **Yes** if proxy is required for the Rest Service. That is, if the Rest API is outside OFSAA. For information on how to enable proxy, see [Enabling Proxy for REST Service Application Rule](#) section.

4.3.9 For Attribute Expression Application Rule

✕
Attribute Builder

Attribute Selection

Name ?

RuleType ? DecisionRule ▼

ExecutionType ? Attribute Expression ▼

Attribute ? FCT_ISSUE_ACTIONV_TEXT_ATTR_5 ▼

Add

Attribute Values

	Name	Value
✕	FCT_ISSUE_ACTIOND_ISSUE_COMPLETION_DATE	
✕	FCT_ISSUE_ACTIONN_OR_STATUS_CD	

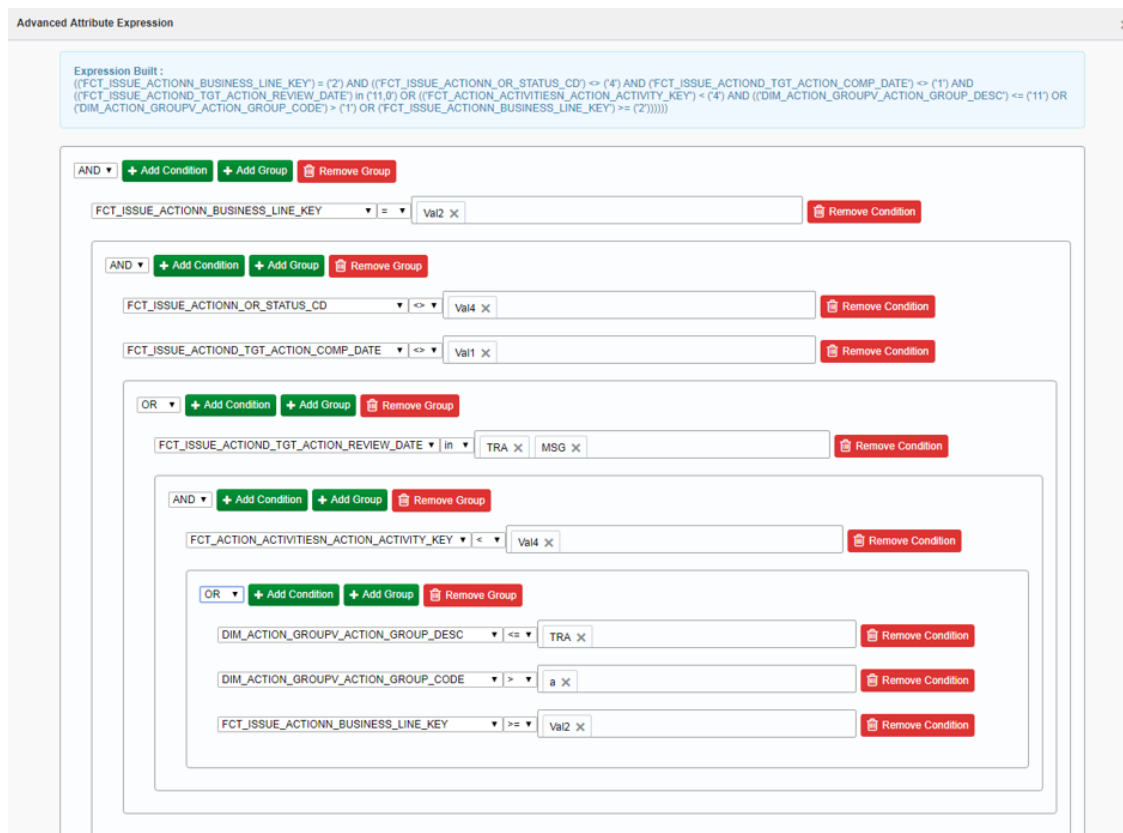
Ok

Field Name	Description
Name	Enter a unique name for the Application Rule.
Rule Type	Displays the rule type as DecisionRule. This Rule Execution type supports only DecisionRule type.
Execution Type	Displays the Application Execution Type as Attribute Expression.
Attribute	<p>Select the attribute for which you want to define the application rule, from the drop-down list. The list displays the attributes configured for the selected application and component. For more information, see Configuring Application Object Model (AOM) section.</p> <p>Click Add to add values to the selected attributes. A row is added in the <i>Attribute Values</i> pane. Click the Value column to select the values for the attribute from the drop-down. You can select one or more values.</p> <p>You can delete a row by clicking button.</p> <p>You can select multiple attributes and click Add to assign values to those attributes.</p>

4.3.10 For Advanced Attribute Expression Application Rule

This is an advanced version of Attribute Expression Application Rule with additional logical conditions and assignment operators. The expression can be dynamically built and will return True or False value after evaluation. This is used as a DecisionRule in transitions.

You can define this application rule with multiple conditions and nested groups.



Field Name	Description
Rule Name	Enter a unique name for the Application Rule.
AND/ OR	Select the logical operator to be used for the conditions in a group.

Field Name	Description
Add Condition	<p>When you click Add Condition, a new row gets added. To define a condition, select the attribute, the operator and the value from the drop-down lists. Multiple values can be selected for each attribute.</p> <ul style="list-style-type: none"> Attribute- The drop-down list displays the attributes configured for the selected application and component. For more information, see Configuring Application Object Model (AOM) section. Operator- Available options are in,=,<>,<,<=,>,>=. Value- Displays the values configured for the selected attributes. Select the required value. <p>Click Remove Condition to delete already added condition.</p>
Add Group	<p>Click Add Group if you want to have nested conditions. For each group, select the required logical operator as AND or OR.</p> <p>Click Remove Group to delete a group of conditions.</p>

4.3.11 For JSON Path Expression Application Rule

This Application rule is used to extract data from the JSON Path Expression, which gets returned from a Rest API or Web Service call, and you can store it into a Data Field for further processing.

Rule Details ✕

Name ?

RuleType ?

ExecutionType ?

JSON input ?

JSON Path Expression ?

Output Datafield ?

Scope ?

Covert to Type ?

Field Name	Description
Name	Enter a unique name for the Application Rule.
Rule Type	<p>Select the rule type from the drop-down list. The available rule types are Decision Rule and Execution Rule.</p> <p>For Decision Rule, output of JSON Path Expression will be compared with RHS expression and rule will return either true or false accordingly.</p> <p>For execution rule, JSON Path Expression is evaluated and the output is returned to the DataField selected as Output DataField.</p>
Execution Type	Displays the Application Execution Type as JSON Path Expression.
JSON Input	Select the Data Field in which the output of Web Service is stored from the drop-down list. You should select a DataField which has JSON as its value.
JSON Path Expression	Enter the JSON path expression. For more information, see JsonPath Expressions section.
Operator	<p>This field is displayed only if Rule Type is selected as Decision Rule.</p> <p>Select the required operator for comparison from the drop-down list. The options are =, <, >, >= and <=.</p>
RHS Expression	<p>This field is displayed only if Rule Type is selected as Decision Rule.</p> <p>Enter the expression to which you want to compare the JSON path expression.</p>
Output DataField	<p>This field is displayed only if Rule Type is selected as Execution Rule.</p> <p>Select the DataField to which you want to return the value of JSON Path Expression, from the drop-down list.</p>
Scope	<p>Select the scope of the Application Rule from the drop-down list. The options are:</p> <ul style="list-style-type: none"> ▪ Process- Select Process if you want to use the Application Rule only in the current process. ▪ Package- Select Package if you want to use the Application Rule across all the processes in the Application package.
Convert To Type	Select JSON ARRAY to store the output in Array format or select String to store as a string, from the drop-down list.

4.3.12 JSON Read From DB Application Rule

This Application Rule is used to read data from the database in JSON format.

JSON Read from DB
✕

Name ? ✕

RuleType ? ExecutionRule ▼

ExecutionType ? JSON Read From DB ▼

Table Name ?

Column List ?

Where Condition ?

Return JSON Type ? JSON Array ▼

Output Datafield ? JSON_DB_DATA ▼

Scope ? Process ▼

Field Name	Description
Name	Enter a unique name for the Application Rule.
Rule Type	Only Execution Rule type is supported.
Execution Type	Displays the Execution Type as JSON Read From DB.
Table Name	Enter the table name from which you want to read the data.
Column List	Enter the column names of the selected table.
Where Condition	Enter the filter condition (where clause) of the SQL query.

Field Name	Description
Return JSON Type	Select the JSON type of the returned value as JSON Object or JSON Array based on your requirement. JSON Object- Select this option if the returned value is a single row. JSON Array- Select this option if the returned data has multiple rows.
Output DataField	Select the DataField to which you want to return the value of the SQL query, from the drop-down list.
Scope	Select the scope of the Application Rule from the drop-down list. The options are: <ul style="list-style-type: none"> ▪ Process- Select Process if you want to use the Application Rule only in the current process. ▪ Package- Select Package if you want to use the Application Rule across all the processes in the Application package.

4.3.13 JSON Write To DB Application Rule

This Application Rule is used to write the data in JSON format into the database.

Rule Details
✕

Name ?

RuleType ? ExecutionRule ▼

ExecutionType ? JSON Write To DB ▼

Table Name ?

Source JSON ?

```
{{"N_COMP_ID":"1","N_COMP_CODE":"1","V_COMP_NAME":"RAO R"}, {"N_COMP_ID":"2","N_COMP_CODE":"2","V_COMP_NAME":"Compliance Investigation"}}
```

Output Datafield ? ▼

Scope ? Process ▼

Field Name	Description
Name	Enter a unique name for the Application Rule.

Field Name	Description
Rule Type	Only Execution Rule type is supported.
Execution Type	Displays the Execution Type as JSON Write To DB.
Table Name	Enter the table name to which you want to write the data in JSON format.
Source JSON	Enter the data in the JSON format that you want to write to the database.
Output DataField	This field is not applicable.
Scope	<p>Select the scope of the Application Rule from the drop-down list. The options are:</p> <ul style="list-style-type: none"> ▪ Process- Select Process if you want to use the Application Rule only in the current process. ▪ Package- Select Package if you want to use the Application Rule across all the processes in the Application package.

4.4 Configuring Application Object Model (AOM)

This module helps in creating set of attributes for a given application in an abstract way, so that frameworks like PMF and other modules can leverage to retrieve application attributes and its values.

Each application is identified using an application package ID. For configuring package IDs, see [Application Package](#) section.

Against each package id, the set of attributes needs to be seeded in the “AAI_AOM_APP_COMP_ATTR_MAPPING” table.

Column Name	Description
APP_COMP_ATTR_MAP_ID	<p>Enter a unique ID for the attribute.</p> <p>You need to enter Attribute name and description for each attribute ID entered here in the AAI_AOM_APP_COMP_ATTR_TL table. See AAI_AOM_APP_COMP_ATTR_TL Table section.</p>
V_ATTR_CODE	Name of the attribute.
N_ATTR_TYPE_ID	<p>ID of the attribute type.</p> <p>The values of the attributes are fetched based on attribute type.</p> <p>1001- Static</p>

Column Name	Description
	1002- SQL Query 1003- JavaAPI 1004- Hierarchy Code For more information, see Attribute Types .
V_ATTRIBUTE_VALUE1 V_ATTRIBUTE_VALUE2	Values to be fetched for the attribute. Based on the attribute type, you need to pass the values.
N_APP_ID	Application code for which the current attribute is configured. For example, if you are configuring Run execution parameters for IFRS application, enter the application ID of IFRS here.
N_COMP_ID	Component code for which the attribute is configured.
V_UDP_CODE	Special property used by applications (user defined). For example, 'GET_STATUS' –to get the status for the workflow.
V_ATTR_CONTROL_TYPE	Enter the Control type ID to be used for the attribute. For example, 3 is used for drop-down list, 7 for textbox, 11 for date control, 41 is for hierarchy.

4.4.1 Attribute Types

The values of attributes are fetched based on the attribute types. Following are the attribute types with their IDs:

- **1001 (Static)** - Store attribute values in the `AAI_AOM_STATIC` table as `V_STATIC_ID` and `V_STATIC_VAL`.
- **1002 (Query)** - Enter the SQL query in `V_ATTRIBUTE_VALUE1` in the `AAI_AOM_APP_COMP_ATTR_MAPPING` table, which has to be fired to fetch the attribute values.
- **1003 (JavaAPI)** – Enter the method that is configured for `V_ATTRIBUTE_VALUE1` for the required attribute. The configured method in the class path is invoked to get the attribute values in this case.
- **1004 (Hierarchy Code)** – Specify the Hierarchy code to be fetched in `V_ATTRIBUTE_VALUE1` in the `AAI_AOM_APP_COMP_ATTR_MAPPING` table.

4.4.2 AAI_AOM_APP_COMP_ATTR_TL Table

In this table, for each `APP_COMP_ATTR_MAP_ID` table, enter the locale specific Attribute Name, Description as shown:

	APP_COMP_ATTR_MAP_ID	V_ATTR_NAME	V_ATTR_DESC	V_LOCALE_CODE
1	134	Legal Entity	Legal Entity	en_US
2	135	Consolidation Type	Consolidation Type	en_US
3	136	Reporting Currency	Reporting Currency	en_US
4	137	Bucket Conversion	Bucket Conversion	en_US
5	138	Interpolation Method	Interpolation Method	en_US
6	139	PD Interpolation Method	PD Interpolation Method	en_US
7	140	Source Run Id	Source Run Id	en_US
8	141	FIC MIS Date	FIC MIS Date	en_US
9	142	Run Execution Description	Run Execution Description	en_US

4.4.3 Usage of AOM Attributes in Run Pipeline

While executing Run pipeline, the Select Run Params window displays the AOM fields which are marked as mandatory.

For example, for the attributes stored in APP_COMP_ATTR_MAP_ID table as shown in the previous figure, the Select Run Params window will be displayed as shown:

The 'Select Run Params' dialog box contains the following fields:

- TASK_STATUS**: Text input field.
- Bucket Conversion**: Dropdown menu with 'Mid of Bucket' selected.
- PD Interpolation Method**: Dropdown menu with 'Non-Linear Geometric' selected.
- Interpolation Method**: Dropdown menu with 'Cubic Spline' selected.
- FIC MIS Date**: Date picker showing 'mm/dd/yy'.
- Consolidation Type**: Dropdown menu with 'Solo' selected.
- Source Run Id**: Text input field.
- Legal Entity**: Text input field.
- Reporting Currency**: Text input field.
- Run Execution Description**: Text input field.

An 'OK' button is located at the bottom center of the dialog.

5 Designing a Pipeline

Business pipelines are defined in OFSAA to design and execute sequence of tasks which are either OFSAA tasks or external tasks, to derive a well-defined outcome. This flow is defined by using various OFSAA artifacts from the component toolbar.

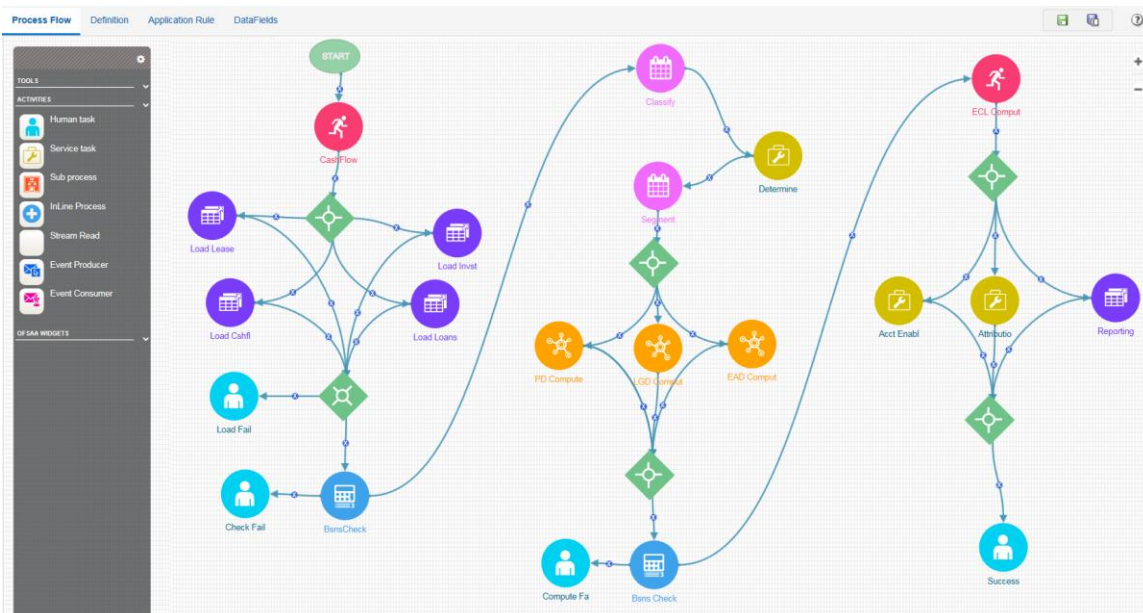
Using Process Modeler, we can:

- a) Orchestrate a Business pipeline
- b) Orchestrate a Run pipeline using PMF modeling

5.1 Orchestration of a Business Pipeline

Business Pipeline is used to design a Business Process which consists of a sequence of tasks either internal or external tasks through well-defined interfaces. Using the designer, you can design the entire business flows consisting of various types of tasks or another business pipeline.

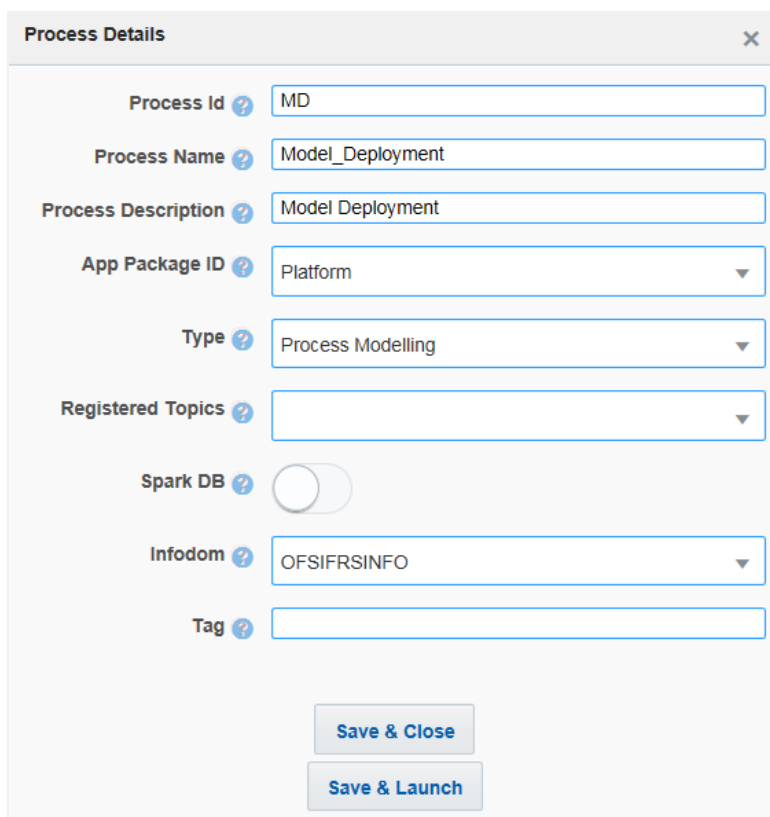
5.1.1 An Example of a Business Pipeline



In the example shown, we use various OFSAA widgets like Run, LoadT2T, RuleType3, TransformDT, MFModel and activities like Human Tasks and Service tasks, which are related to each other through transitions. For executing tasks in parallel, we used Parallel Gateways.

5.1.2 Creating a Business Pipeline

Click  in the *Process Modeller* Summary window.

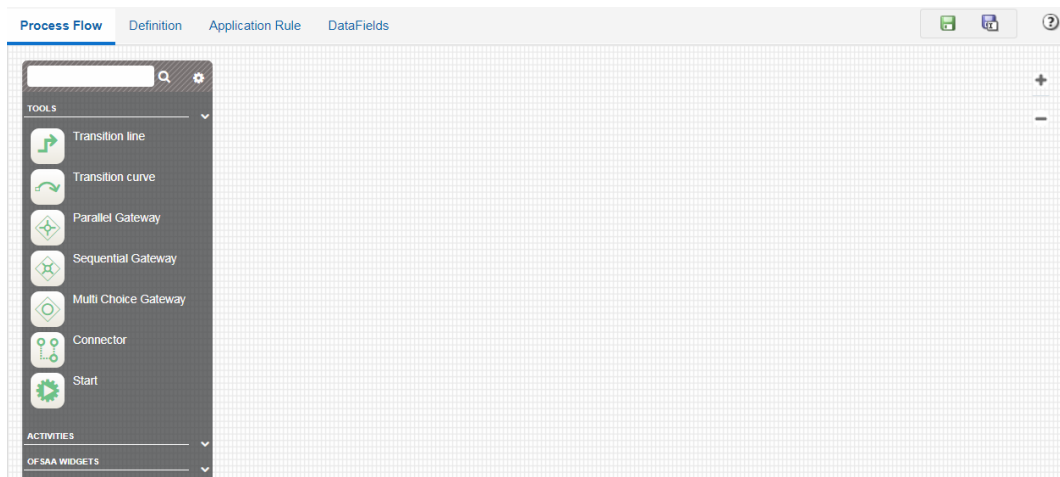



The screenshot shows a window titled "Process Details" with a close button (X) in the top right corner. The form contains the following fields and controls:

- Process Id**: Text input field containing "MD".
- Process Name**: Text input field containing "Model_Deployment".
- Process Description**: Text input field containing "Model Deployment".
- App Package ID**: Drop-down menu showing "Platform".
- Type**: Drop-down menu showing "Process Modelling".
- Registered Topics**: Drop-down menu (empty).
- Spark DB**: Toggle switch (turned off).
- Infodom**: Drop-down menu showing "OFSIFRSINFO".
- Tag**: Text input field (empty).

At the bottom of the form, there are two buttons: "Save & Close" and "Save & Launch".

1. Enter a unique Process ID, Process Name and a description.
2. Select the appropriate app package in which you want to create the process. For more information, see [Application Package](#) section.
3. Select **Process Modelling** from the **Type** drop-down list.
4. Select the information domain in which you want to create the Business Pipeline, from the **Infodom** drop-down list. The list displays all the infodoms mapped to the applications configured in your OFSAA instance.
5. Click **Save & Launch**. The Process Flow canvas is displayed.

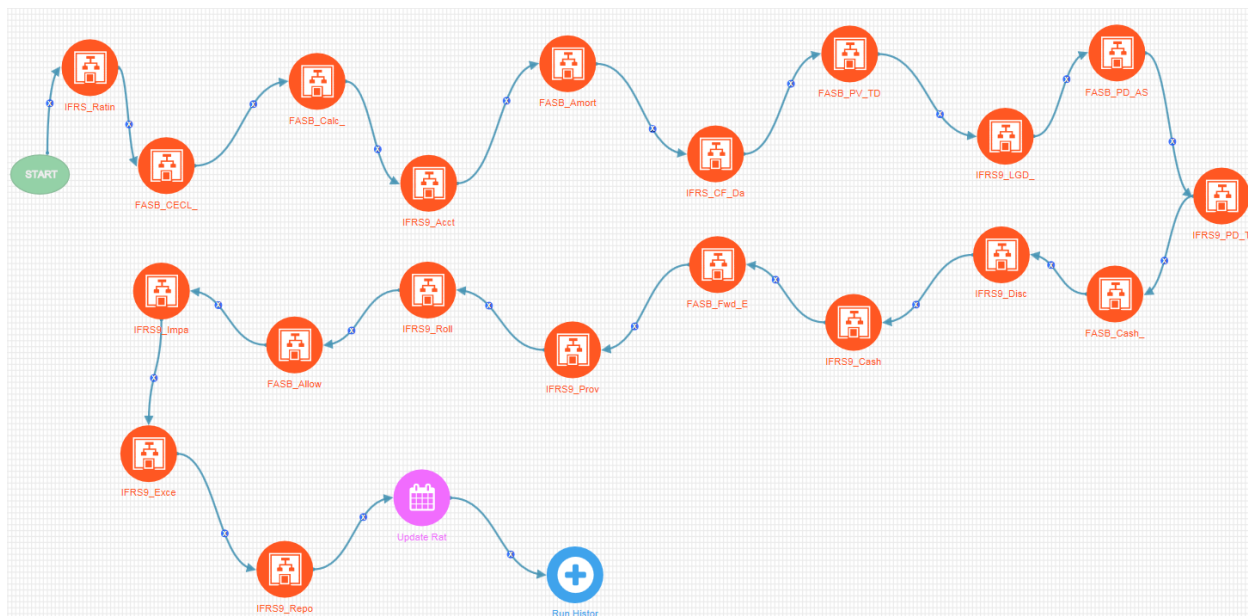


6. Click  **Start** from the *Tools* pane in the floating toolbar and click on the drawing canvas where you want to place it. This Start activity indicates the beginning of the Process.
7. Design your Process with various components available in the *Process Flow* tab. For more information on each component, see [Components for Designing Your Process Flow](#) section.


5.2 Run Pipeline

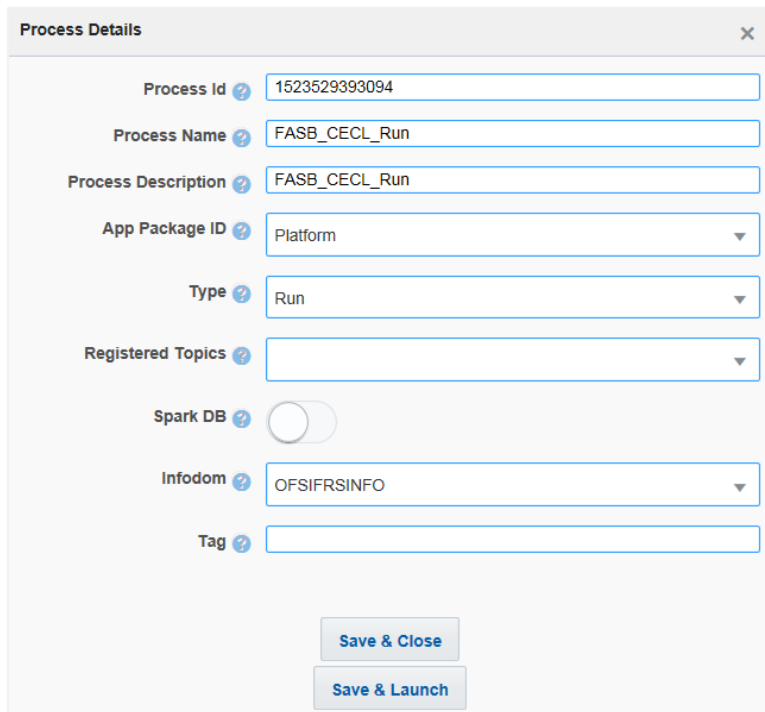
Run Process is used to create a Run definition in Rule Run Framework (RRF) using PMF Process. Visual representation of the Run is enabled through PMF by construction of a Run Pipeline. Various OFSAA widgets that enables construction of Run Pipeline are available in the Component toolbar.

5.2.1 An Example of Run Pipeline



5.2.2 Creating a Run Pipeline

1. Seed AOM Data Fields. For more information, see [Configuring Application Object Model \(AOM\)](#) section.
2. Click  in the *Process Modeller* Summary window.

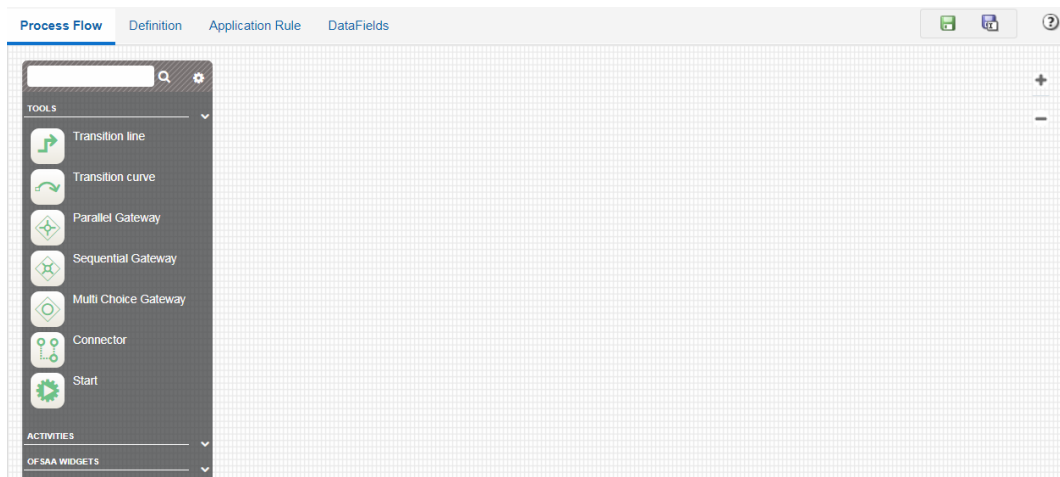



The screenshot shows a 'Process Details' dialog box with the following fields and values:

- Process Id: 1523529393094
- Process Name: FASB_CECL_Run
- Process Description: FASB_CECL_Run
- App Package ID: Platform
- Type: Run
- Registered Topics: (empty)
- Spark DB: (toggle switch)
- Infodom: OFSIFRSINFO
- Tag: (empty)

Buttons at the bottom: Save & Close, Save & Launch

3. Enter a unique Process ID, Process Name and a description.
4. Select the appropriate app package in which you want to create the process. For more information, see [Application Package](#) section.
5. Select **Run** from the **Type** drop-down list.
6. Select the information domain in which you want to create the Run Process, from the **Infodom** drop-down list. The list displays all the infodoms mapped to the applications configured in your OFSAA instance.
7. Click **Save & Launch**.



8. Click  **Start** from the *Tools* pane in the floating toolbar and click on the drawing canvas where you want to place it. This Start activity indicates the beginning of the Process.

5.2.3 Design your Run Pipeline using OFSAA Widgets

You can construct Run pipeline using only OFSAA Widgets. For details, see [Configuring OFSAA Tasks in Your Process Flow](#) section.

5.2.4 Design your Run Pipeline using Sub Pipeline

You can construct Run Pipeline using already constructed pipelines. The same pipeline can be used across different Run pipelines. For more information, see [Calling another Pipeline from Your Parent Pipeline](#) section.


5.2.5 Design your Run Pipeline using Inline Process

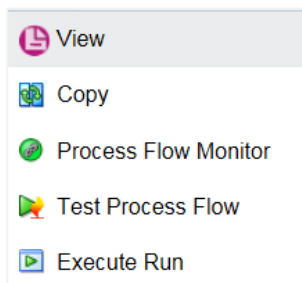
You can construct Run Pipeline by grouping a set of tasks as an Inline Process. For more information, see [Inline Process](#) section.

5.2.6 Design your Run Pipeline using combinations of OFSAA Widgets, Sub Pipeline and Inline Process

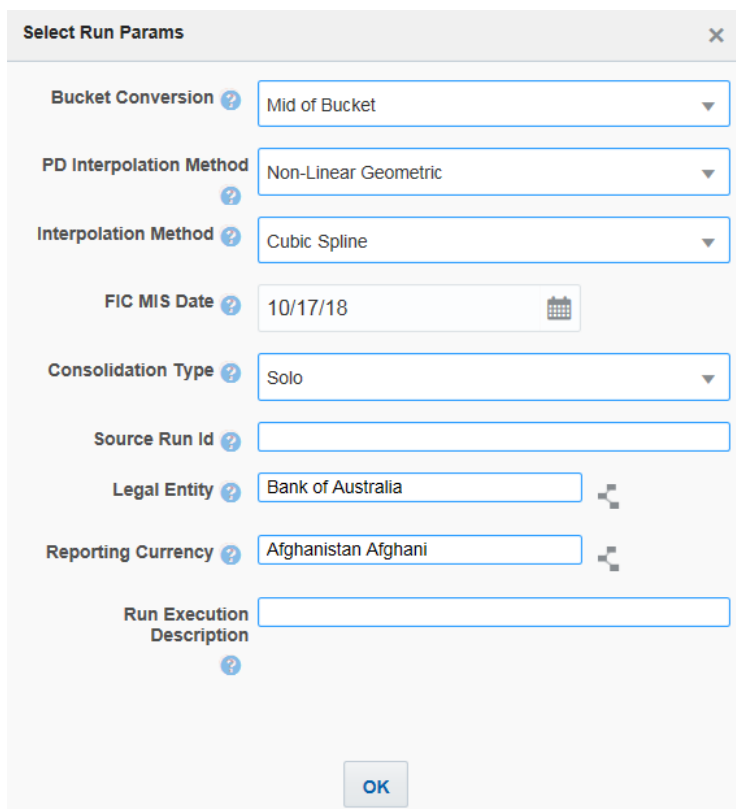
You can construct Run pipeline by combining OFSAA widgets, Sub pipeline and Inline Process.

5.2.7 Executing Run Pipeline

1. From the *Process Modeller Summary* window, click  corresponding to the Run Pipeline you want to execute:



2. Click **Execute Run** to execute the Run Pipeline. The *Select Run Params* window is displayed.

A dialog box titled 'Select Run Params' with a close button (X) in the top right corner. The dialog contains several fields for configuration:

- Bucket Conversion**: A dropdown menu with 'Mid of Bucket' selected.
- PD Interpolation Method**: A dropdown menu with 'Non-Linear Geometric' selected.
- Interpolation Method**: A dropdown menu with 'Cubic Spline' selected.
- FIC MIS Date**: A date input field showing '10/17/18' with a calendar icon to its right.
- Consolidation Type**: A dropdown menu with 'Solo' selected.
- Source Run Id**: An empty text input field.
- Legal Entity**: A text input field containing 'Bank of Australia' with a refresh icon to its right.
- Reporting Currency**: A text input field containing 'Afghanistan Afghani' with a refresh icon to its right.
- Run Execution Description**: An empty text input field with a help icon (?) below it.

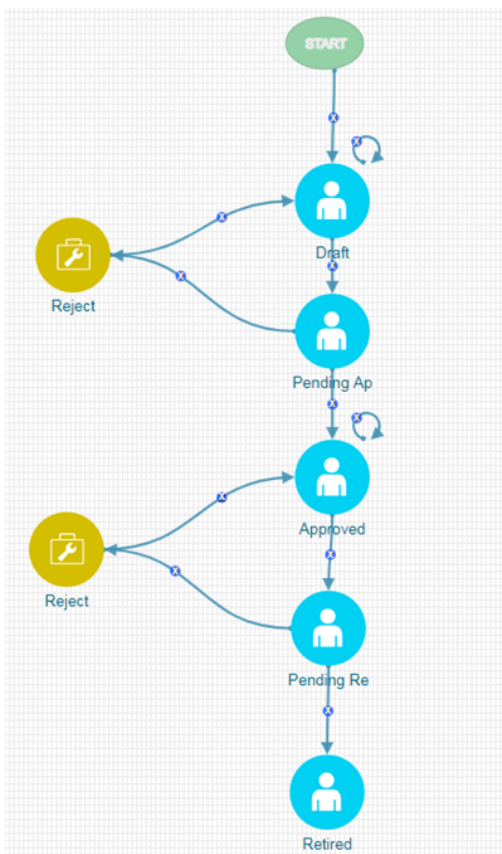
An 'OK' button is located at the bottom center of the dialog.

3. Select values for the Run Parameters and click **OK**.



The execution of the Run Pipeline is triggered using the selected FIC MIS DATE. The RUNSKEY is generated and inserted into the "DIM_RUN" table. For the RUNSKEY generated, the corresponding user selected parameters are inserted in to the "RUN_EXE_PARAMETERS" table.

6 Human Tasks

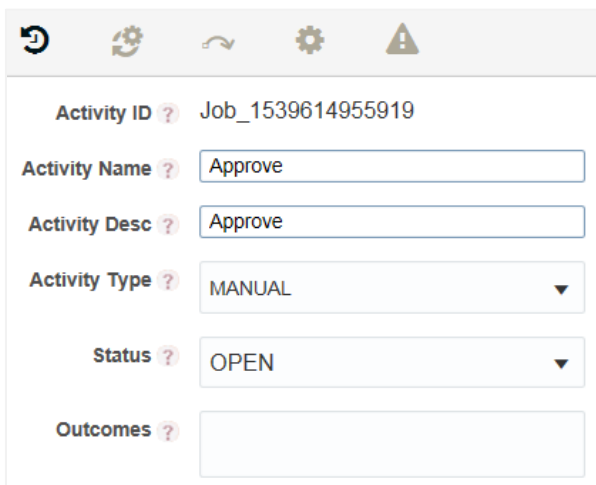
Human Task is used if an activity requires a human intervention to go to the next activity.



6.1 How to Use Human Task

1. From the *Process Flow* tab, click  Human Task from the *Activities* pane in the floating toolbar and click on the drawing canvas where you want to place it.
2. Double-click  icon. The *Activity* tab is displayed.

6.1.1 Activity Tab




The screenshot shows a form with the following fields and values:

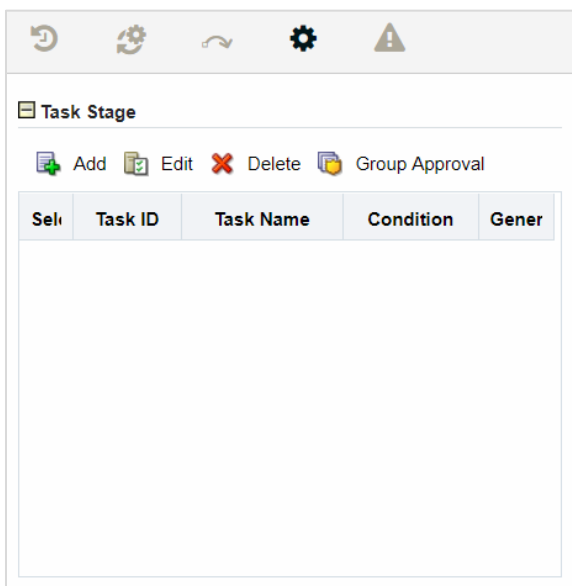
Field	Value
Activity ID	Job_1539614955919
Activity Name	Approve
Activity Desc	Approve
Activity Type	MANUAL
Status	OPEN
Outcomes	

1. **Activity ID** is auto populated.
2. Enter Activity Name and Description.
3. Select the **Status** of the activity from the drop-down list. The list displays the seeded values in the AAI_WF_STATUS_B table.

6.1.2 Action Tab for Creating Tasks/Notification

Action/Task is used to inform the assigned user about an action to be completed in the current stage of Workflow. You can add multiple tasks for an activity. Task can be assigned to a user, user group, users with a particular role, or users selected by an application rule. Task is sent to the Application Inbox of the assigned users.

1. Click  to define Actions.



2. Click  **Add** from the *Action* tab to add a new Task.

6.1.2.1 Defining Task Details

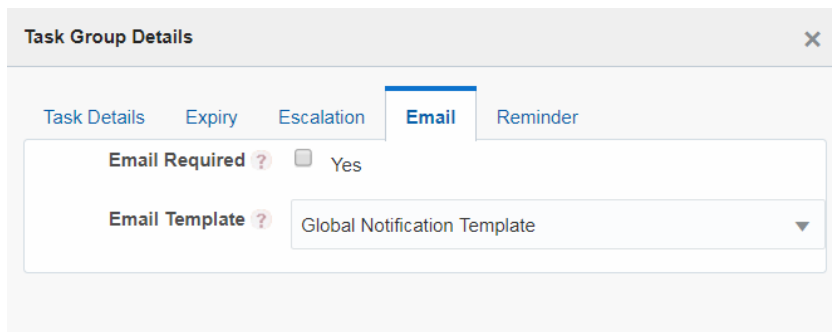
The screenshot shows the 'Task Group Details' dialog box with the 'Task Details' tab selected. The dialog has a close button (X) in the top right corner. Below the tab are five sub-tabs: 'Task Details', 'Expiry', 'Escalation', 'Email', and 'Reminder'. The 'Task Details' sub-tab is active and contains the following fields:

- Task ID** ? : 1513333455171
- Task Name** ? : TaskName_
- Condition** ? : Default
- Generate** ? : On Entry
- Enabled** ? : Yes

1. Task ID is automatically generated.
2. Enter a Task Name. By default, task name is prefixed with "TaskName_".
3. Select the decision rule so that when the condition is satisfied the Task will be sent to the users associated to this task.
4. Select the required option for Generate:
 - On Entry- Task is sent before executing the activity business logic.
 - On Exit- Task is sent before leaving the activity and moving to next activity.
 - On Stage- When there is a transition where the target and source is the same activity, and if that transition happens, then OnStage tasks are triggered.

5. Select the **Yes** checkbox to enable the Task.

6.1.2.2 Setting Email Notification



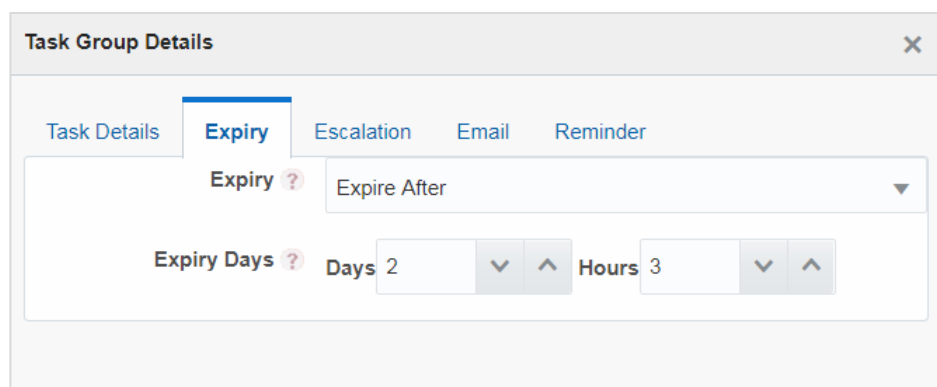
The screenshot shows a dialog box titled "Task Group Details" with a close button (X) in the top right corner. It has five tabs: "Task Details", "Expiry", "Escalation", "Email", and "Reminder". The "Email" tab is currently selected. Inside the dialog, there are two main sections. The first section is labeled "Email Required" with a question mark icon and a checkbox that is checked, followed by the text "Yes". The second section is labeled "Email Template" with a question mark icon and a dropdown menu that currently displays "Global Notification Template".

You should have configured Email settings before you set email notification for your tasks. For more details on how to configure email settings, see [Configuring Email for Human Tasks](#) section.

1. Select the **Yes** checkbox if an email needs to be sent for the task.
2. Select the email template to be used from the drop-down list. Templates are populated from `AAI_WF_Email_Template` table.

6.2 Additional Functionalities for Human Tasks

6.2.1 Setting Task Expiry

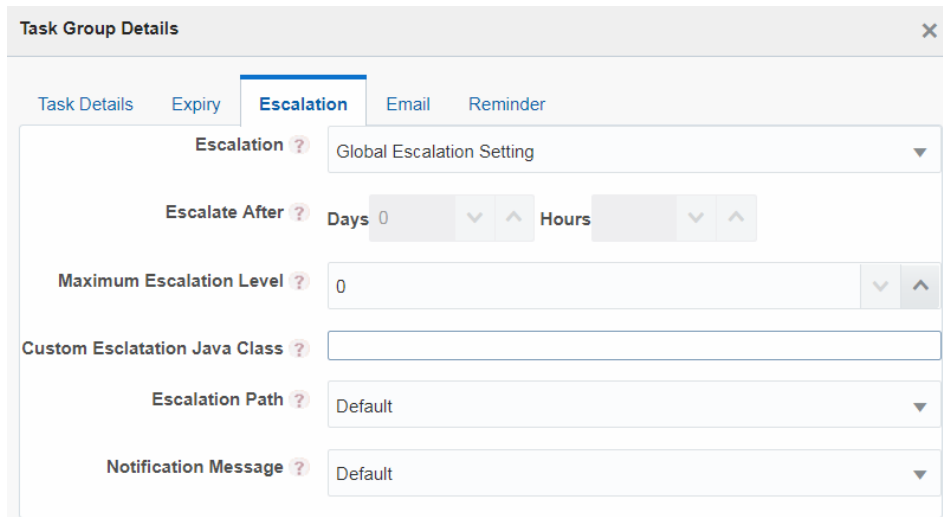


The screenshot shows the same "Task Group Details" dialog box, but now the "Expiry" tab is selected. The "Expiry" dropdown menu is open, showing "Expire After" as the selected option. Below this, there are two input fields: "Expiry Days" with a value of "2" and "Hours" with a value of "3". Each input field has up and down arrow buttons for adjustment.

1. Select the required option from the **Expiry** drop-down list:
 - Global Expiry Setting- Select this option to set the task expiry based on global setting. Global setting can be set from backend.
 - Never Expire- Select this option if the task should not expire.
 - Expire After- Select this option if you want to set the task expiry after some days and/or hours.
 - Dynamic Value – Select this option if you want the user assigned to the task to set the task expiry date and time dynamically. This value needs to be entered in the code.

2. If **Expiry** is selected as **Expire After**, enter the number of days and/or hours after which the task should be expired.

6.2.2 Setting Task Escalation



The screenshot shows the 'Task Group Details' dialog box with the 'Escalation' tab selected. The dialog contains the following fields:

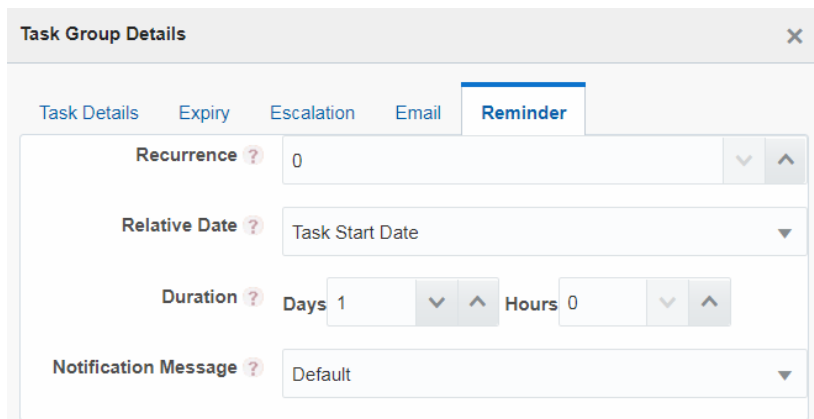
- Escalation**: A dropdown menu set to 'Global Escalation Setting'.
- Escalate After**: Two spinners for 'Days' (set to 0) and 'Hours'.
- Maximum Escalation Level**: A spinner set to 0.
- Custom Escalation Java Class**: An empty text input field.
- Escalation Path**: A dropdown menu set to 'Default'.
- Notification Message**: A dropdown menu set to 'Default'.

1. Select the **Escalation** criteria:
 - Global Escalation Setting- Select this option to set the task escalation based on global setting. Global setting can be set from backend.
 - Never Escalate - Select this option if escalation is not required for the task.
 - Escalate After- Select this option if you want to escalate if the task is not addressed after some days and/or hours.

NOTE: Escalation should be set after expiry of the task. If you have selected **Never Expire** option for **Expiry**, you cannot set escalation.

2. If **Escalate After** is selected for **Escalation**, enter the number of days and/or hours after which the escalation should be triggered.
3. Enter the maximum number of escalation level. 1 indicates escalation to the immediate manager. 2 indicates escalation to manager's manager and so on.
4. Enter the custom escalation Java Class which you want to call.
5. Select the escalation path from the drop-down list. The options are Default, People Hierarchy and Custom Rule.
6. Select the type of notification message from the drop-down list. Notification messages are populated from `AAI_WF_Templates` table.

6.2.3 Setting Reminder for Your Task



The screenshot shows a 'Task Group Details' dialog box with a 'Reminder' tab selected. The dialog contains the following fields:

- Recurrence**: A numeric input field with the value '0' and up/down arrows.
- Relative Date**: A dropdown menu with 'Task Start Date' selected.
- Duration**: Two input fields for 'Days' (value '1') and 'Hours' (value '0'), each with up/down arrows.
- Notification Message**: A dropdown menu with 'Default' selected.

Reminders will be sent to the assigned user as an open task in their inbox.

1. Enter the number of times you want to set the reminder from the **Recurrence** field.
2. Select **Task Start Date** if you want to send reminder after the defined number of days and/or hours, from the start date of the task. Select **Task Expiration Date** if you want to send reminder before the defined number of days and/or hours from the end date/expiry date of the task.
3. Select the number of days and/or hours from the **Relative Date** after/before which you want to set the reminder.
4. Select the **Notification Message** you want to send, from the drop-down list. Notification messages are populated from `AAI_WF_Templates` table.
5. Click **OK**.

7 Service Tasks



Service Task is an automatic task which gets triggered in the Process flow. It is used to execute the Business Logic which is defined through Application Rule of Execution Rule type. For more information on configuring Application Rule, see [Application Rule](#) section.

Service Tasks are typically used to invoke External Model Service through Rest API, External Java APIs, Stored Procedures, and Functions.

More detailed explanation of invoking external model service is available in [How to invoke External Model through Web Service](#) section.



7.1 How to Use a Service Task

1. From the *Process Flow* tab, click  **Service Task** from the *Activities* pane in the floating toolbar and click on the drawing canvas where you want to place it.
2. Double-click  icon. The *Activity* tab is displayed.

7.1.1 Activity Tab

1. **Activity ID** is auto populated.
2. Enter Activity Name and Activity Description.

7.1.2 Implementation Tab

1. Click . The *Implementation* tab is displayed.

The screenshot shows the configuration interface for a Rule. It is organized into three sections:

- Rule:** Contains an 'Infodom' dropdown menu with the value 'OFSALMINFO' and an 'Execution Rule' dropdown menu with the value 'Balance Sheet Load Ch'.
- Parameters:** Includes an 'Add' button and two parameter entries. The first entry is 'WF_OBJECT_ID' with a search icon and a red 'X' icon. The second entry is 'SUM_BALANCES' with a search icon and a red 'X' icon.
- Pre/Post Processing:** Contains 'Pre Rule' and 'Post Rule' dropdown menus, both set to 'Default'.



2. Select the information domain where the Execution Rule you want to execute is available, from the **Infodom** drop-down list. The list displays all the Infodoms mapped to the applications configured in your OFSAA instance.
3. Select the **Execution Rule** which needs to be executed for this activity. For more information on how to define an Application Rule, see [Application Rules](#) section.
 - Click . The *Participant Details* window is displayed with all Application Rules of Execution Rule types available in your Process.
 - Click the Name link of the Application Rule to view the details.
 - Select the required Rule and click **Ok**.
4. Add Parameters you want to pass to the Execution Rule using Data Fields. You can pass Static values or Dynamic Values. In case of Dynamic, the value needs to be entered during execution of the workflow.
 - Click **Add** under **Parameters**. The *Binding Details* window is displayed.



The screenshot shows the 'Binding Details' dialog box. It contains the following fields:

- Data Field:** A dropdown menu with the value 'SUM_BALANCES'.
- Parameter Type:** A dropdown menu with the value 'Dynamic'.
- Value:** An empty text input field.

An 'Ok' button is located at the bottom center of the dialog.

- Select the **Data Field** to which you want to pass the value. The list displays all Data Fields for the current process or package.
- Select the **Parameter Type** as Static to pass a static value to the selected Data Field in the **Value** field or as Dynamic to pass the value during execution of the workflow.

The added parameters are displayed under *Parameters* pane. You can click  corresponding to a parameter to edit it or click  to delete it.

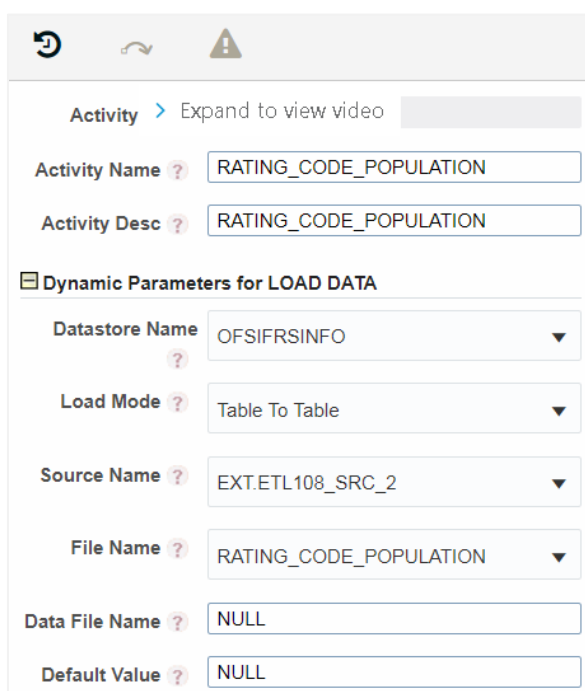
5. Select an application rule which you want to execute before executing the Execution Rule.
 - Click  corresponding to **Pre Rule** and select the required application rule.
6. Select an application rule which you want to execute after executing the Execution Rule.
 - Click  corresponding to **Post Rule** and select the required application rule.

8 Configuring OFSAA Tasks in Your Process Flow

OFSAA widgets are used to orchestrate OFSAA components like T2T definitions, PLC definitions (DT), Rules (Classification Rule and Computation Rule), Models in EMF, Run, and Data Quality Groups into your Process Flow using Process Modeller.


8.1 How to Use an OFSAA Widget

1. From the *Process Flow* tab, click the required widget under **OFSAA Widget** toolbar in the left pane and click the canvas. The available OFSAA Widgets are LoadT2T, TransformDT, RuleType2, RuleType3, MFModel, Run, and DataQualityGroups.
2. Double-click the widget. On the Right Pane, the *Activity* tab is displayed.



The screenshot shows the configuration window for an OFSAA widget. At the top, there are three icons: a refresh icon, a undo icon, and a warning icon. Below the icons, the 'Activity' tab is selected, and the text 'Expand to view video' is visible. The configuration fields are as follows:


- Activity Name:** RATING_CODE_POPULATION
- Activity Desc:** RATING_CODE_POPULATION
- Dynamic Parameters for LOAD DATA:**
 - Datastore Name:** OFSIFRSINFO
 - Load Mode:** Table To Table
 - Source Name:** EXT.ETL108_SRC_2
 - File Name:** RATING_CODE_POPULATION
 - Data File Name:** NULL
 - Default Value:** NULL


3. Enter the Activity Name and Activity Description.
4. Based on the OFSAA widget, the Dynamic Parameters are displayed. For more information, see [Dynamic Parameters for OFSAA Widgets](#) section.
5. To view the definition of the OFSAA component you are using, mouse over its icon and click  View. The Definition window of the OFSAA component is displayed.


8.1.1 Dynamic Parameters for OFSAA Widgets


8.1.1.1 LoadT2T

Dynamic Parameters for LOAD DATA



Datastore Name  ▼

Load Mode  ▼

Source Name  ▼

File Name  ▼


Data File Address


 **Default Value** 


Field Name	Description
Datastore Name	Select the Information Domain in which the Data Mapping you want to execute is present, from the drop-down list.
Load Mode	Table to Table should be selected for Data Mapping definitions such as T2T, T2H, H2T, H2H and L2H definitions.
Source Name	Select the required source on which the Data Mapping definition you want to execute is defined, from the drop-down list.
File Name	Select the Data Mapping definition you want to execute, from the drop-down list. Based on the selected Load Mode and Source Name , the list displays the corresponding definitions.
Data File Address	This field is not applicable to Load Mode selected as Table to Table.
Default Value	Select the Data Fields you want to pass as parameters for the selected Data Mapping definition. For information on creating Data Fields, see Data Fields section. For additional information, see <i>Task Component Parameters</i> section in OFS Analytical Applications Infrastructure User Guide .

8.1.1.2 TransformDT 

Dynamic Parameters for TRANSFORM DATA

Datstore Name 


Rule Name 


Parameter List 


Field Name	Description
Datstore Name	Select the Information Domain in which the PLC you want to execute is present, from the drop-down list.
Rule Name	Select the Post Load Changes (DT) definition you want to execute from the drop-down list. The list displays the <i>Post Load Changes</i> definitions in the selected Information Domain.
Parameter List	Select the Data Fields you want to pass as parameters for the selected Data Mapping definition. For information on creating Data Fields, see Data Fields section.


8.1.1.3 RuleType2  or RuleType3 

Dynamic Parameters for RULE_EXECUTION

Datstore Name 

Rule Code 

Build Flag 


Optional Parameters 


Field Name	Description
Datstore Name	Select the Information Domain in which the RRF Rule you want to execute is present, from the drop-down list.


Field Name	Description
Rule Code	<p>Display the codes of the RRF Rules defined under the selected Infodomain.</p> <p>Select the required Rule from the drop-down list.</p>
Build Flag	<p>Select the required option from the drop-down list as “Yes” or “No”.</p> <p>Build Flag refers to the pre-compiled rules, which are executed with the query stored in database. While defining a Rule, you can make use of Build Flag to fasten the Rule execution process by making use of existing technical metadata details wherein the rule query is not rebuilt again during Rule execution.</p> <p>Built Flag status set to “No” indicates that the query statement is formed dynamically retrieving the technical metadata details. If the Build Flag status is set to “Yes” then the relevant metadata details required to form the rule query is stored in database on “Save” of a Rule definition. When this rule is executed, database is accessed to form the rule query based on stored metadata details, thus ensuring performance enhancement during Rule execution. For more information, see <i>Significance of Pre-Built Flag</i> section in OFS Analytical Applications Infrastructure User Guide.</p>
Optional Parameters	<p>Select the Data Fields you want to pass as parameters for the selected Data Mapping definition.</p> <p>For information on creating Data Fields, see Data Fields section.</p>


8.1.1.4 MFModel

Dynamic Parameters for MODEL

Datstore Name 

Model Code 

Operation 

Optional Parameters 

Field Name	Description
Datstore Name	Select the Information Domain in which the RRF Rule you want to execute is present, from the drop-down list.
Model Code	Display the codes of the EMF Models defined under the selected Infodom.
Operation	The All definition for the Operation field conveys the process of extracting the data from the flat files and applying the run regression on the data extracted. For Batches that are being built for the first time the data will be extracted from the flat files and the run regression will be applied on it.
Optional Parameters	Select the Data Fields you want to pass as parameters for the selected Data Mapping definition. For information on creating Data Fields, see Data Fields section.

8.1.1.5 Run

Dynamic Parameters for RUN

Infodom ?

Run Type ?

Execution Rule ?

Run Parameters ?

Field Name	Description
Infodom	
Run Type	Select Base Run or Simulation Run based on the type of the Run you want to execute, from the drop-down list.
Execution Rule	Select the Run you want to execute from the drop-down list.
Run Parameters	Select the Data Fields you want to pass as parameters for the selected Data Mapping definition. For information on creating Data Fields, see Data Fields section.

8.1.1.6 DataQualityGroups 

Dynamic Parameters for RUN DQ RULE

Datstore Name ▼
OFSALMINFO

User Id AAAIUSER

DQ Group Name ▼
LRMDQGRP22

Rejection Threshold
[]

Additional Parameters
[]

Fail if Threshold Breaches ▼
No

Optional Parameters
-1

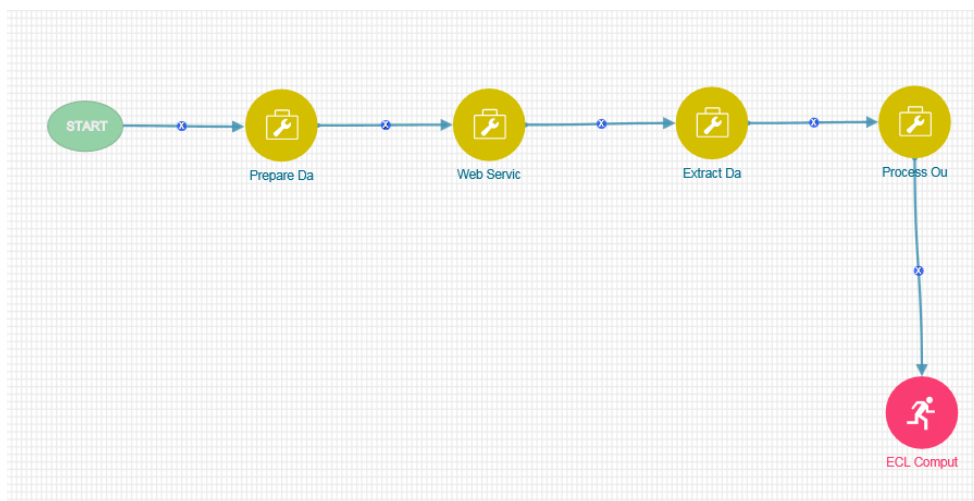
Property	Description
DQ Group Name	Refers to the Data Quality Groups consisting of associated Data Quality Rule definition(s). Select the required DQ Group from the drop-down list.
Rejection Threshold	Specify the percentage of Rejection Threshold (%) limit in numeric value. This refers to the maximum percentage of records that can be rejected in a job. If the percentage of failed records exceeds the Rejection Threshold, the job will fail. If the field is left blank, the default the value is set to 100%.
Additional Parameters	Specify the Additional Parameters as filtering criteria for execution in the pattern Key#Data type#Value; Key#Data type#Value;...etc. Here the Data type of the value should be "V" for Varchar/Char, or "D" for Date with "MM/DD/YYYY" format, or "N" for numeric data. For example, if you want to filter some specific region codes, you can specify the Additional Parameters value as \$REGION_CODE#V#US;\$CREATION_DATE#D#07/06/1983;\$ACCOUNT_BAL#N#10000.50; Note: In case the Additional Parameters are not specified, the default value is fetched from the corresponding table in configuration schema for execution.

Property	Description
Fail if Threshold Breaches	<p>Select Yes or No from the drop-down list. If Yes is selected, execution of the task fails if the threshold value is breached. If No is selected, execution of the task continues.</p> <p>Note: For Custom Check type DQ Rules in Hive Infodoms, the execution of the task will not fail even if the threshold is breached. This is a limitation.</p>
Parameters	<p>Select the Data Fields you want to pass as parameters for the selected Data Mapping definition.</p> <p>For information on creating Data Fields, see Data Fields section.</p>

9 Orchestrating External Models/Components in Your Process Flow

You can use external models or external components in your process flow by using Rest Service Application Rule or External Java API Application Rule.

9.1 How to invoke External Model through Web Service



A typical External Web Service invocation will have three steps:

1. Data Preparation
2. Web Service Invocation
3. Data Extraction

9.1.1 Data Preparation

Before invoking Web Service, we need to prepare data to be passed across to the Web Service. This can be done by configuring Application Rule.

In the above example, data preparation is done through JSON Read From DB Application Rule as shown below, which reads data from a table, converts into JSON and stores output to JSON_DB_DATA Data Field.

The screenshot shows a dialog box titled "Edit API Details" with a close button (X) in the top right corner. The dialog contains several fields and dropdown menus:


- Name**: Read DB Data
- RuleType**: ExecutionRule
- ExecutionType**: JSON Read From DB
- Table Name**: DIM_ACCOUNT
- Column List**: N_ACCT_SKEY Account_ID,v_account_desc Account_Name
- Where Condition**: V_PROD_CODE='CME'
- Return JSON Type**: JSON Array
- Output Datafield**: JSON_DB_DATA (highlighted with a red box)
- Scope**: Process

A "Save" button is located at the bottom center of the dialog.

For more information, see [JSON Read From DB Application Rule](#) section.

9.1.2 Webservice Invocation

Consider you want to use an external model called ForecastModelPost, which is available in `http://<IP Address/hostname of the Web Server>:<servlet port>/<context name>/rest-api/v1/ModelWebService/ForecastModelPost`.

1. Launch the Process for which you want to add an external model.
2. Go to *Application Rule* tab and click  **Add >Rest Service**.

Rule Details [X]

Name ? ModelWebService

Rule Type ? ExecutionRule

Execution Type ? Rest Service

Method Type ? POST

Url ? http://[redacted]/ofsaai/rest-api/v1/ModelWebService/ForecastModelPost

Query Param ?

Headers ?


Data ? {"owner": {"id": 3730}, "scenario_name": "Baseline", "forecast_periods": 2, "data": ~~JSON_DB_DATA~~}

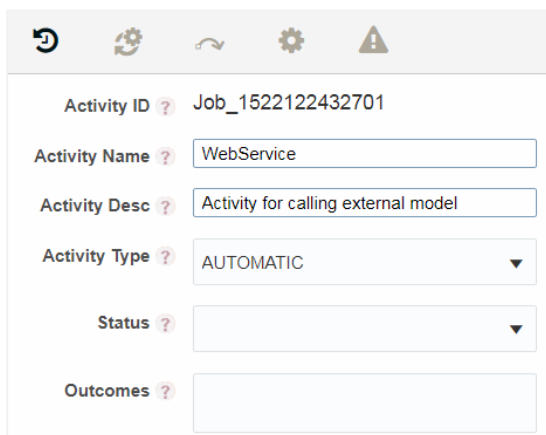
Return Parameter ? TASK_RESPONSE

Scope ? Process

Is Proxy Required ? Yes

Save

3. Enter the details as shown. For more information on configuring a Rest Service, see [Rest Service Application Rule](#) section.
4. In this example, the Data sent to the webservice is a combination of both static and dynamic value (~~JSON_DB_DATA~~ is data field holding a JSON string which is prepared as previous step in the pipeline.)
5. Click **Save**. The ModelWebService Application Rule gets created in your logged in Information Domain.
6. Go to *Process Flow* tab.
7. Click  **Service Task** from the *Activities* pane and click on the drawing canvas where you want to place it.
8. Double-click the Service Task icon.



Activity ID ? Job_1522122432701



Activity Name ?

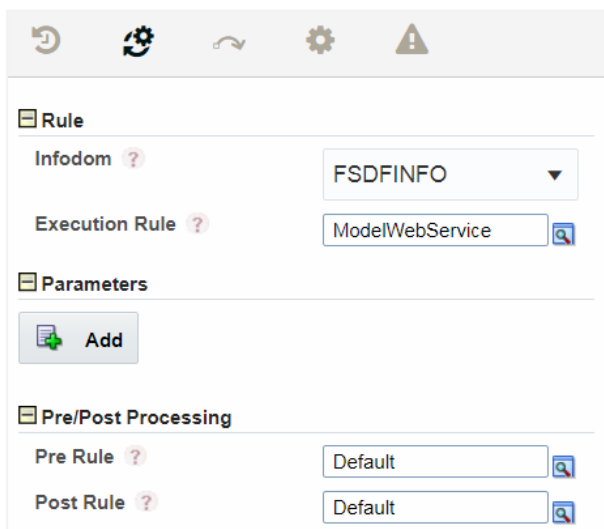
Activity Desc ?

Activity Type ?

Status ?

Outcomes ?

9. Enter the Activity details such as Activity Name and Activity Description.
10. Click . The *Implementation* tab is displayed.
11. Select the information domain where the ModelWebService Application Rule is defined, from the **Infodom** drop-down list. The list displays all the Infodoms mapped to the applications configured in your OFSAA instance.
12. Click  in the **Execution Rule** field. The *Participant Details* window is displayed with all Application Rules of Execution Rule types available in your Process.
13. Select ModelWebService and click **Ok**.



Rule

Infodom ?

Execution Rule ?

Parameters

Pre/Post Processing

Pre Rule ?

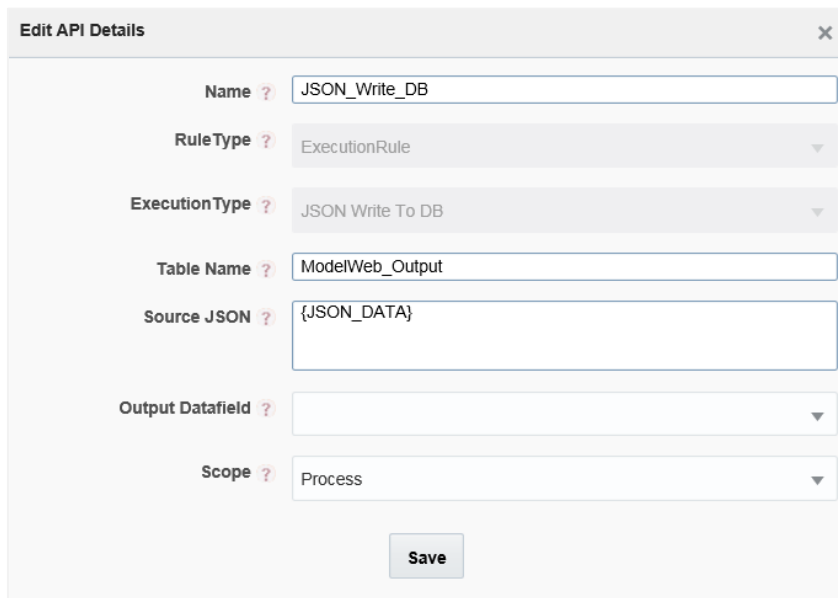
Post Rule ?

9.1.3 Data Extraction

The response from the Web Service needs to be processed depending on the application requirement. PMF has capabilities to process the JSON and store in the output table.

You can use JSON PATH expressions to extract the relevant information from the Web Service response. For more information on JSON PATH expressions, see [For JSON Path Expression Application Rule](#) section.

Similarly, the response can be stored back to the database using JSON Write To DB Application Rule as shown:



The screenshot shows a dialog box titled "Edit API Details" with a close button (X) in the top right corner. The dialog contains several fields and dropdown menus:

- Name**: A text input field containing "JSON_Write_DB".
- RuleType**: A dropdown menu with "ExecutionRule" selected.
- ExecutionType**: A dropdown menu with "JSON Write To DB" selected.
- Table Name**: A text input field containing "ModelWeb_Output".
- Source JSON**: A text input field containing "{JSON_DATA}".
- Output Datafield**: A dropdown menu that is currently empty.
- Scope**: A dropdown menu with "Process" selected.

At the bottom center of the dialog is a "Save" button.

For more information, see [JSON Write To DB Application Rule](#) section.

10 Configuring Custom Components

An entry into AAI_WF_COMPONENT_REGISTRATION table will create a new custom component in the PMF OFSAA widgets.

Row 1	Fields
V_COMPONENT_ID	ATTRIBUTIONCOMPONENT
V_COMPONENT_EXE_IMPL	com.ofs.aai.service.wf.ofsaa.component.AttrTaskImpl
V_SYNCHRONOUS_FLAG	N
V_COMPONENT_UI_LAUNCH	attribution/attributiondetail.jsp
V_COMPONENT_PARENT_ID	
V_COMPONENT_NAME	ATTRIBUTION ANALYSIS COMPONENT
V_COMPONENT_LIST_CLASS	com.ofs.aai.pr2.comp.impl.AttributionList
V_COMPONENT_TYPE	I
V_COMPONENT_CSS	
V_COMPONENT_ICONS	{{"id":"","iName":"Add","iImg":"action.svg","iUrl":"/dataQuality/DQRuleServlet?pageMode=NEW

V_COMPONENT_ID – Specify a unique ID for the component.

V_COMPONENT_EXE_IMPL - This is the interface which needs to be implemented by the component owner, so that during the execution this component, this API is invoked.

V_COMPONENT_UI_LAUNCH - This is the URL of the page which needs to be opened when the component is double clicked in the canvas.

V_COMPONENT_NAME – Specify a name for the component, which will be displayed for the custom component in the OFSAA Widgets.

V_COMPONENT_ICONS - All the icons and menu are configured in this field.

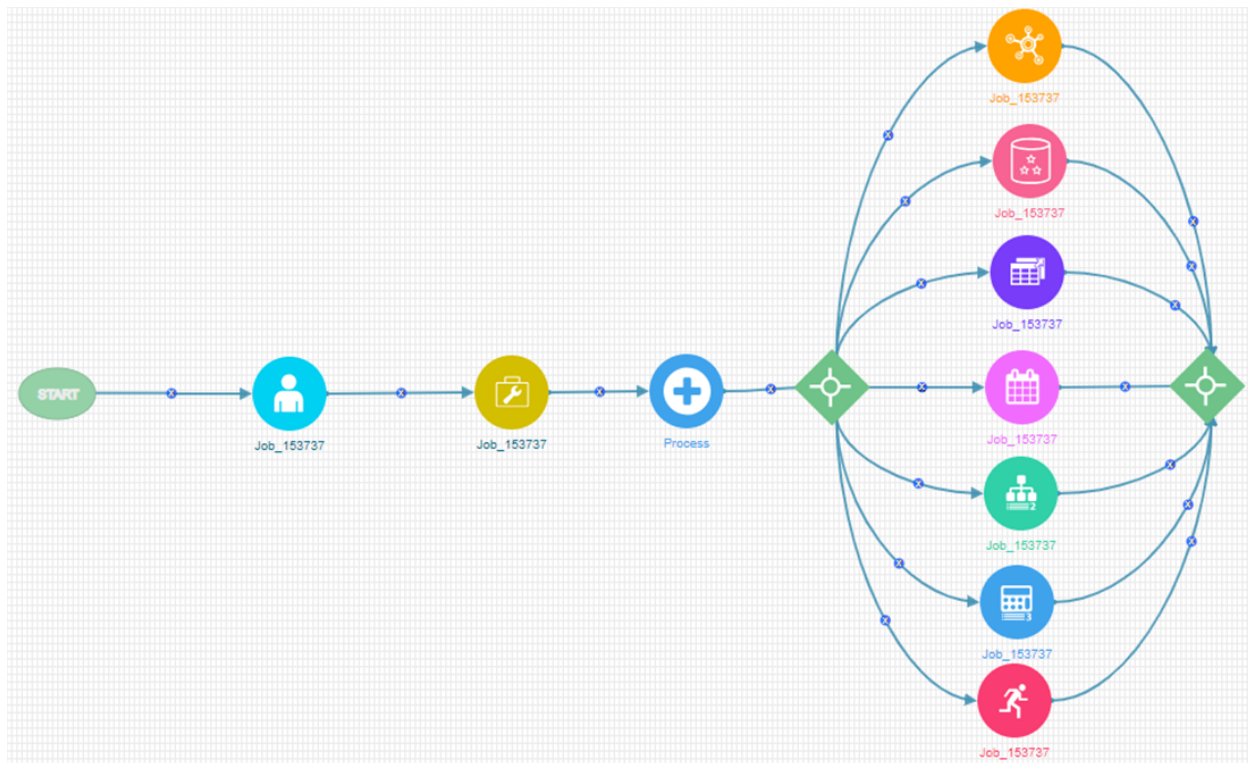
For example,




```

V_COMPONENT_ICONS [{"id":"","iName":"Add","iImg":"action.svg","iUrl":"/dataQuality/DQRuleServlet?pageMode=NEW|
Large Data Editor - aai_wf_component_registration.v_component_icons
Text RTF XML HTML Hex External
[{"id":"","iName":"Add","iImg":"action.svg","iUrl":"/dataQuality/DQRuleServlet?pageMode=NEW| chr(38) ||actionType=
GET_DQ_GROUP_SCREEN|| chr(38) ||infodom={Datastore Name}"},
{"id":"","iName":"Edit","iImg":"edit.svg","iUrl":"/dataQuality/DQRuleServlet?dqGroupId={DQ Group Name}|| chr(38) ||pageMode=
EDIT|| chr(38) ||infodom={Datastore Name}|| chr(38) ||actionType=GET_DQ_GROUP_SCREEN"},
{"id":"","iName":"View","iImg":"previewTree.svg","iUrl":"/dataQuality/DQRuleServlet?dqGroupId={DQ Group Name}|| chr(38) ||
pageMode=VIEW|| chr(38) ||infodom={Datastore Name}|| chr(38) ||actionType=GET_DQ_GROUP_SCREEN"},
{"id":"","iName":"Delete","iImg":"delete.svg","iUrl":""}]
    
```

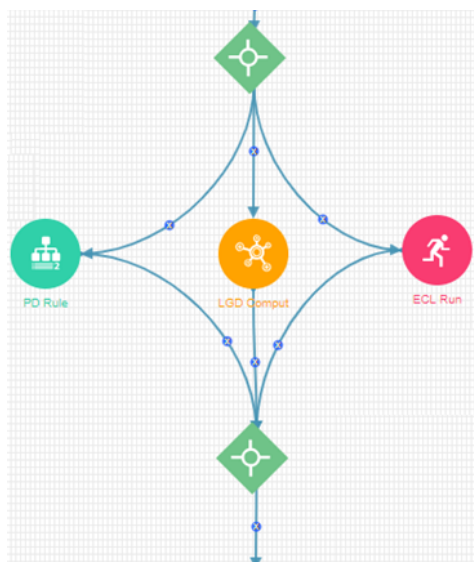

11 Executing Parallel Tasks

Parallel Gateway is used to execute multiple tasks in parallel. In the usual flow, tasks are executed sequentially.







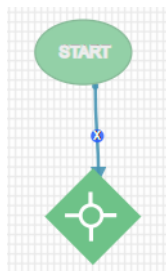
In the shown example, after  is executed, Parallel Gateway  gets invoked. All the OFSAA components which are placed between Parallel Gateways  are executed simultaneously. It waits till all components are executed, and then moves to the next activity in the Process Flow.

11.1 How to Use Parallel Gateways





This section explains how to design 3 tasks such as RuleType 2, MFModel and Run widget to be executed in parallel.

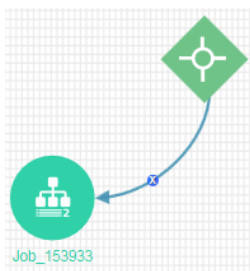
1. Create a Process Modelling Process.
2. Launch the Process. It displays the *Process Flow* tab.
3. Click  **Start** from the *Tools* pane in the floating toolbar and click on the drawing canvas where you want to place it. This Start activity indicates the beginning of the Process.
4. Click  Parallel Gateway from the *Tools* pane in the floating toolbar and click on the drawing canvas where you want to place it.
5. Click  **Transition Curve** and click **Start** (from where the transition starts) and click  (the activity to which you want to have the transition).






6. Click  **RuleType2** from the *OFSAW Widgets* pane, and then click the drawing canvas where you want to place it.

7. Click  **Transition Curve** from the *Tools* pane and click  Parallel Gateway and then

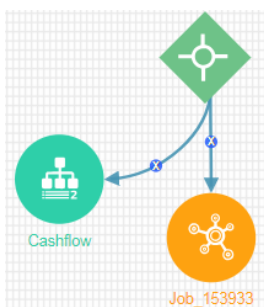

click  .




8. Configure the RuleType2 widget. For more information, see [Configuring OFSAA Tasks in Your Process Flow](#) section.
9. Click  **MFModel** from the *OFSAA Widgets* pane, and then click the drawing canvas where you want to place it.



10. Click  **Transition Curve** from the *Tools* pane and click  Parallel Gateway and then


click  .

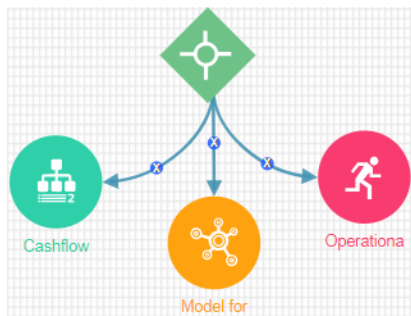


11. Configure the MFModel widget. For more information, see [Configuring OFSAA Tasks in Your Process Flow](#) section.


12. Click  **Run** from the *OFSAA Widgets* pane, and then click the drawing canvas where you want to place it.


13. Click  **Transition Curve** from the *Tools* pane and click  Parallel Gateway and then

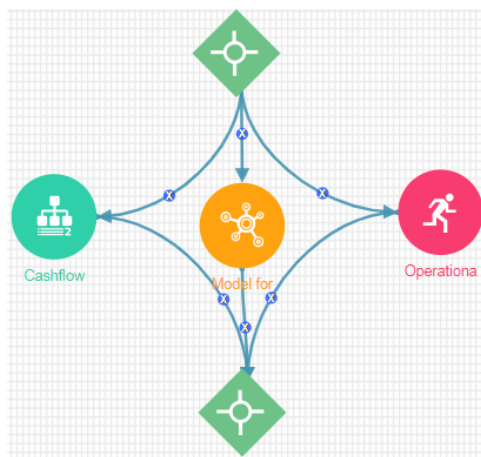

click .



14. Configure the Run widget. For more information, see [Configuring OFSAA Tasks in Your Process Flow](#) section.

15. Click  Parallel Gateway from the *Tools* pane in the floating toolbar and click on the drawing canvas where you want to place it.

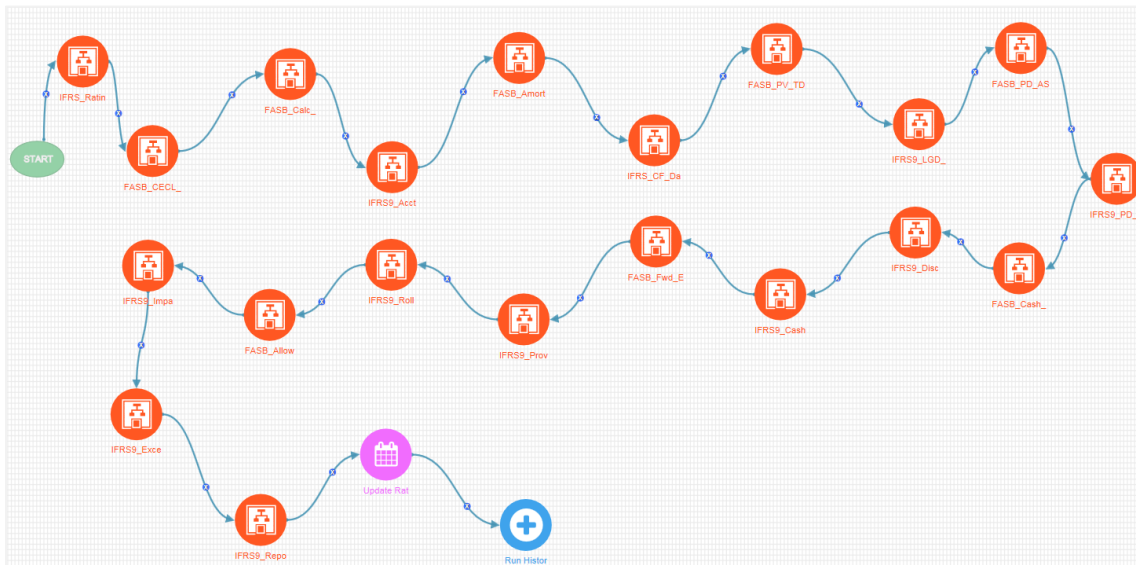
16. Draw transitions from RuleType2, MFModel and Run widgets to the merging  Parallel Gateway as shown:




NOTE: You can drag and position the widgets to avoid overlapping of widgets or transition lines.

12 Calling another Pipeline from Your Parent Pipeline

Reusability is important while designing your pipeline. Sub Pipeline is the mechanism in the PMF to call another pipeline from your parent pipeline.



12.1 How to Configure Sub Pipeline

1. From the *Process Flow* tab, click  **Sub Pipeline** under *Activities* toolbar in the left pane and click the drawing canvas where you want to place it.



2. Double-click . The *Sub Process Details* window is displayed.

Sub Process Details

Activity Name ?

Activity Desc ?

App Package ID ?

Process ID ?

Object ID ?

Object Type ?

Data field pass ? Yes

Calling Mode ? Synchronous Asynchronous

3. Enter the details as given in the table:

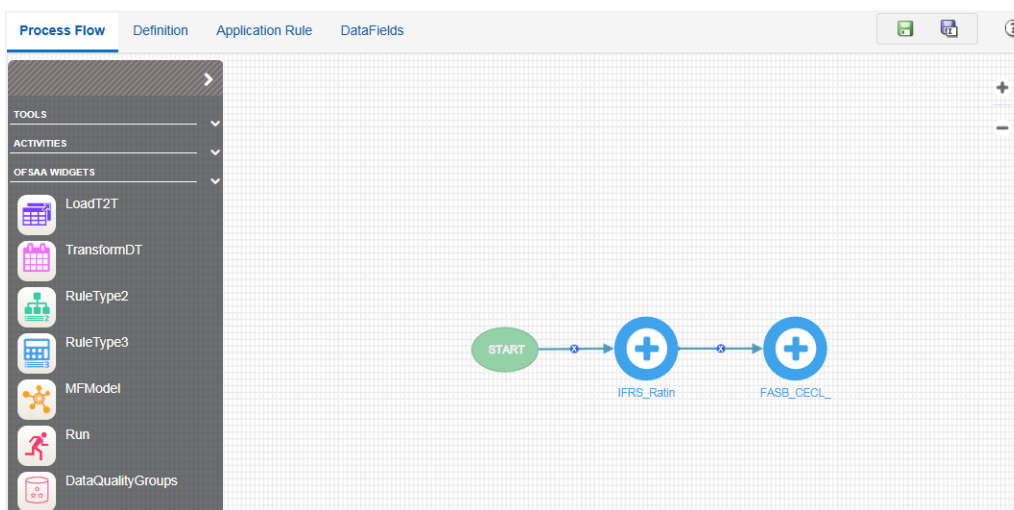
Field Name	Description
App Package ID	Select the Application Package from which you want to call a process, from the drop-down list. The package IDs that are seeded from backend are displayed in the list.
Process ID	Select the Process which you want to call within your workflow, from the drop-down list. The list displays all processes defined for the selected Application Package.
Object ID	Select the Object ID from the drop-down list. Object ID is the entity ID used to identify if a workflow needs to be started from the beginning or from the current stage. Object ID drop-down list is populated from the Data Fields.
Object Type	Select the object type from the drop-down list. Object Type is used to identify a workflow that is passed by the application.
Data field pass	Select Yes to pass the parameters passed to the parent workflow to be passed to the selected sub pipeline.
Calling Mode	<ul style="list-style-type: none"> ▪ Synchronous – Select this option if you want the sub pipeline to complete before the flow moves to the next activity. ▪ Asynchronous – Select this option if you do not want to wait for the sub pipeline to complete. And the parent workflow proceeds to the next activity.

13 Inline Process

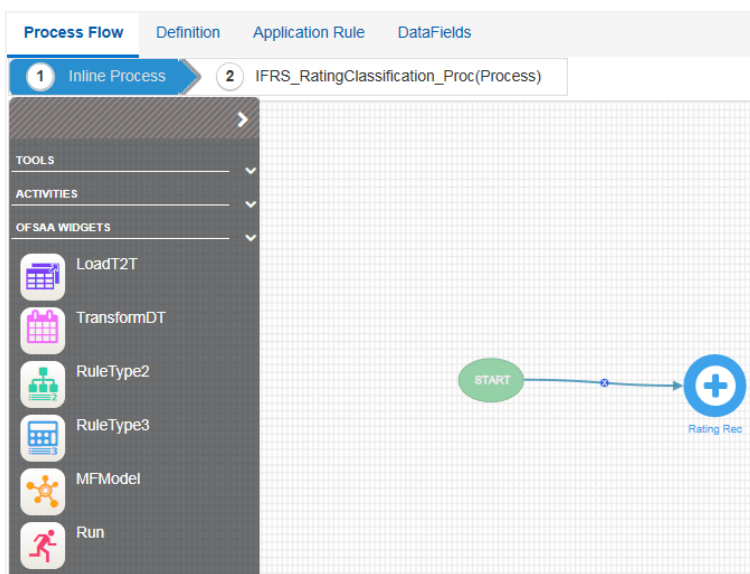
Inline Process is mostly used for readability and usability. It is used to group a set of tasks as an Inline Process. This helps to represent a Process Flow with so many tasks /OFSAAs widgets in a systematic and organized way. You can have Inline Process inside another Inline Process.

For example, if an RRF Run definition has multiple sub processes (RRF), each sub process can be represented as an InLine Process.

13.1 An Example of Inline Process

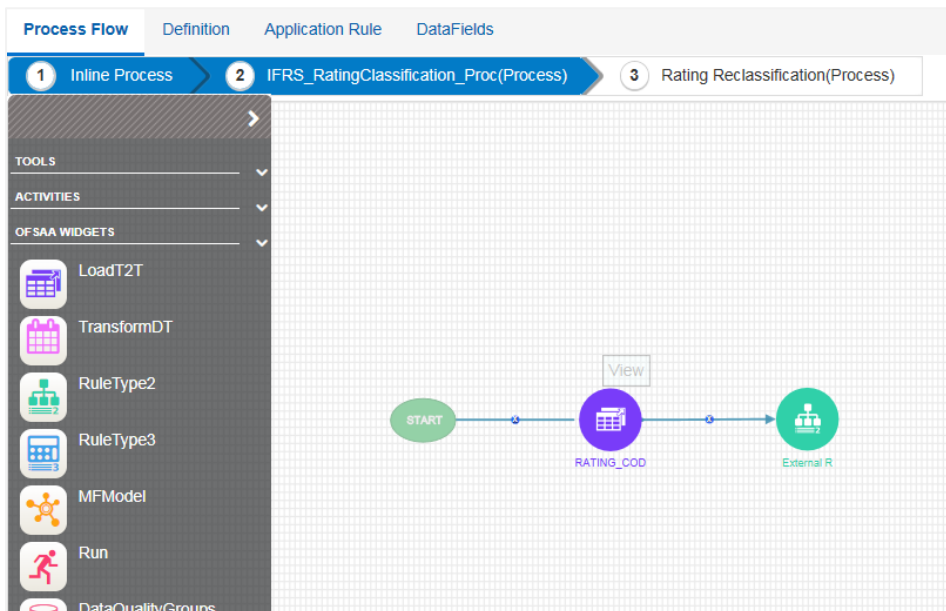


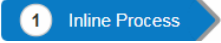
Click [IFRS_Ratin](#) icon. It opens up the process flow diagram of the IFRS_Rating_Classification_Proc as shown:







In this example, IFRS_Rating_Classification_Proc Inline Process has an Inline Process named

Rating Reclassification. Double-clicking  opens up its process flow as shown:




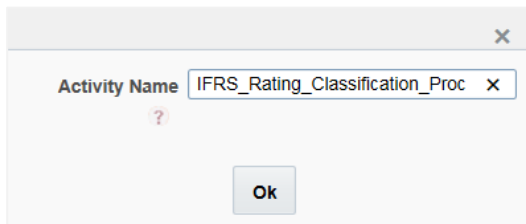
Multiple levels of Inline Process is supported. You can go to any level by clicking the Inline Process name displayed in the header. Click  to go to the parent process flow.

13.2 How to Add an Inline Process

1. Launch the Process to which you want to add an Inline Process. It displays the *Process Flow* tab.
2. Click  **Start** from the *Tools* pane in the floating toolbar and click on the drawing canvas where you want to place it. This Start activity indicates the beginning of the Process.
3. Expand *Activities* floating toolbar and click  **InLine Process**.
4. Click the drawing canvas where you want to place it.
5. Click  **Transition Curve** and click **Start** (from where the transition starts) and click  (the activity to which you want to connect).
6. Mouse over the icon. The sub menu is displayed.



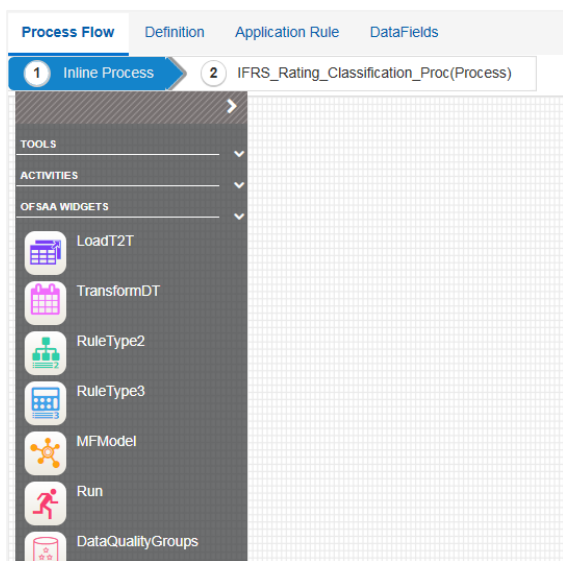
- Click  to modify the Inline Process name.



- Click **Ok**.



- Double-click the Inline Process icon .



- Design the process flow using various components available in the *Process Flow* tab. For more information on each component, see [Components for Designing Your Process Flow](#) section.

14 Configuring Email for Human Tasks

Following table entries should be made for setting Email notification:

14.1 AAI_EMAIL_CONFIG Table

This table holds the SMTP server configurations.

V_PROTOCOL - SMTP

V_HOST –SMTP/ Mail Server ID

V_PORT - SMTP Server Port

V_AUTHENTICATION - Either False or True

V_USER_NAME - Login name to SMTP/ Mail Server ID from which mail will be triggered. This is required if V_AUTHENTICATION is set as True.

V_PASSWORD - Password to login into SMTP/ Mail Server. This is required if V_AUTHENTICATION is set as True.

V_SECURITY –



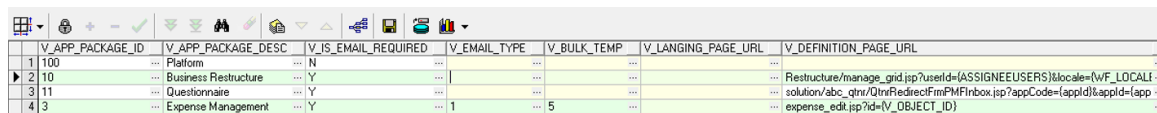
	V_PROTOCOL	V_HOST	V_PORT	V_AUTHENTICATION	V_USER_NAME	V_PASSWORD	V_SECURITY
1	smtp	internal-mail-router.oracle.com	25	false			

14.2 AAI_WF_APP_PACKAGE_B Table

The AAI_WF_APP_PACKAGE_B table entry is for configuring email at Application Level.

- Set the V_EMAIL_REQUIRED parameter value to Y in AAI_WF_APP_PACKAGE_B table .
- Set V_EMAIL_TYPE as:
 - 0 – To receive no notification mails
 - 1 – To get mails instantly
 - 2 – To get bulk mail (Additionally, you need to set V_BULK_MAIL_TRIGGER value to Y in the AAI_WF_BULK_MAIL_TRIGGER table). A single mail will be sent with all the pending notifications from last trigger, as a PDF attachment. Once the bulk mail is sent, V_BULK_MAIL_TRIGGER value is automatically set to N.
 - 3 – To get mail with attachment

V_BULK_TEMP – This is used to set the template for the bulk emails. You can keep this as blank if bulk email is not set.



	V_APP_PACKAGE_ID	V_APP_PACKAGE_DESC	V_IS_EMAIL_REQUIRED	V_EMAIL_TYPE	V_BULK_TEMP	V_LANGING_PAGE_URL	V_DEFINITION_PAGE_URL
1	100	Platform	N				
2	10	Business Restructure	Y				Restructure/manage_grid.jsp?userId={ASSIGNEEUSERS}&locale={VF_LOCALI...
3	11	Questionnaire	Y				solution/abc_qtrn/QttrRedirectFromPMFInbox.jsp?appCode={appld}&appld={app...
4	3	Expense Management	Y	1	5		expense_edit.jsp?id={V_OBJECT_ID}

14.3 AAI_WF_APP_REGISTRATION Table

This table holds email configuration at module or entity type level.

Set the V_IS_EMAIL_REQUIRED parameter value to Y in AAI_WF_APP_REGISTRATION table.

	V_APP_PACKAGE_ID	V_OBJECT_TYPE	V_IS_EMAIL_REQUIRED
1	10	BR	
2	11	QTNR	
3	100	MD_1	
4	3	1000	Y

14.4 AAI_WF_ACTIVITY_TASK_B Table

The AAI_WF_ACTIVITY_TASK_B table holds email configuration at each activity (Task & notification) level.

Set the V_EMAIL_REQUIRED parameter value to Y against the task.

Additionally, you can set the V_EMAIL_TEMPLATE value based on AAI_WF_EMAIL_TEMPLATE table.

	V_PROCESS_ID	V_ACTIVITY_ID	V_TASK_ID	V_CONDITION_TYPE	V_EMAIL_REQUIRED	V_EMAIL_TEMPLATE
1	New	Job_1539062920201	1539062927787	2	Y	4
2	BR1	Activity_1430138133131	1430304038218			
3	BR1	Activity_1430138133131	1430304084815			
4	QTNR	Job_1468916517232	1496309128751	2	Y	2
5	QTNR	Job_1468916574725	1496295700502		Y	2
6	QTNR	Job_1496226571444	1496296080165	2	Y	2
7	QTNR	Job_1496226679422	1496309653931	2	Y	3
8	MD_1	Job_1472554718819	1472554888526	2	N	1

14.5 AAI_USER_PREFERENCE Table

In this table, you can set the user preference of how to receive the notification mails.

V_USER_ID	N_EMAIL_NOTIF_REQ
USER1	1
USER2	2

- 0 – To receive no notification mails
- 1 – To get mails instantly
- 2 – To get bulk mail (Additionally, you need to set V_BULK_MAIL_TRIGGER value to Y in the AAI_WF_BULK_MAIL_TRIGGER table). A single mail will be sent with all the pending notifications from last trigger, as a PDF attachment. Once the bulk mail is sent, V_BULK_MAIL_TRIGGER value is automatically set to N.
- 3 – To get mail with attachment

14.6 AAI_WF_EMAIL_TEMPLATE Table

This table is used to provide the template for the email which needs to be sent.

	V_MAIL_FROM	V_MAIL_MESSAGE	V_MAIL_SUBJECT	V_APP_PACKAGE_ID	V_MAIL_TYPE	N_TEMPLATE_ID
1	workflow_test@oracle.com	<html> <body> <p> Hi {USERID}. </p> <p> This is to inform you that a [TASK_TY ...	[TASK_TYPE]{ENTITYTYPE}{ENTITYID}{TITLE}	0	Notification	1
2	workflow_test@oracle.com	<html> <body> <p> Hi {USERID}. </p> <p> This is to inform you that a [TASK_TY ...	[TASK_TYPE]{ENTITYTYPE}{ENTITYID}{TITLE}	0	Task	4
3	workflow_test@oracle.com	<html> <body> <p> Hi {USERID}. </p> <p> This is to inform you that this a mail dig ...	[ENTITYTYPE]NAME] Mail Digest	0	BulkTask	5
4	workflow_test@oracle.com	<html> <body> <p> Hi {USERID}. </p> <p> This is to inform you that tasks are del ...	Delegation Notification	0	Delegation	2

14.7 AAI_WF_BULK_MAIL_TRIGGER Table

If you have set N_EMAIL_NOTIF_REQ parameter to 2 in AAI_USER_PREFERENCE table, additionally you need to set V_BULK_MAIL_TRIGGER value to Y in the AAI_WF_BULK_MAIL_TRIGGER table.

14.8 CSSMS_USR_PROFILE Table

This table is used to store the email id of the users, to which the notification mails need to be sent.

V_USR_ID	V_EMAIL
USER1	user1@oracle.com
USER2	user2@oracle.com

14.9 AAI_MAIL_AUDIT_TRAIL Table

This is where all mail trails are stored. The status changes once mail is sent. Can be used for debugging purposes.

15 Process Monitor

The Process Monitor is used to monitor current stage of the Process for different instances. After integration with an Application the workflow can be invoked. After invoking it goes through all the stages defined. Process Monitor shows all the stages finished, current stage and stages to come if any. Your user group should be mapped to the function role WFMACC (Workflow Monitor Access) to access Process Monitor window.

Entity ID	Entity Name	Process Name	Process Description	Execution Start Time	Last Execution Time	Last Updated By	Status
200061	200061	Approval Cycle	Approval Cycle for all services	08-AUG-18 11:48:25	13-AUG-18 11:21:24	AAAUSER	RUNNING
DEF_ID-1_2018-08-14-06-17-33	N/A	Monte Carlo Run Execution	Monte Carlo Run Execution	13-AUG-18 11:17:33	13-AUG-18 11:20:47	MRMUSER	RUNNING
DEF_ID-1_2018-08-14-06-19-52	N/A	Portfolio Risk Run Execution	Portfolio Risk Run Execution	13-AUG-18 11:19:52	13-AUG-18 11:20:47	MRUSER	RUNNING
DEF_ID-2_2018-08-09-06-23-32	N/A	Instrument Valuation Run Execution	Instrument Valuation Run Execution	08-AUG-18 11:23:32	08-AUG-18 11:26:31	MRUSER	RUNNING
DEF_ID-2_2018-08-09-06-12-51	N/A	Instrument Valuation Run Execution	Instrument Valuation Run Execution	08-AUG-18 11:12:51	08-AUG-18 11:12:51	AAAUSER	RUNNING
2	2	Approval Cycle	Approval Cycle for all services	08-AUG-18 10:35:59	08-AUG-18 10:35:59	AAAUSER	RUNNING

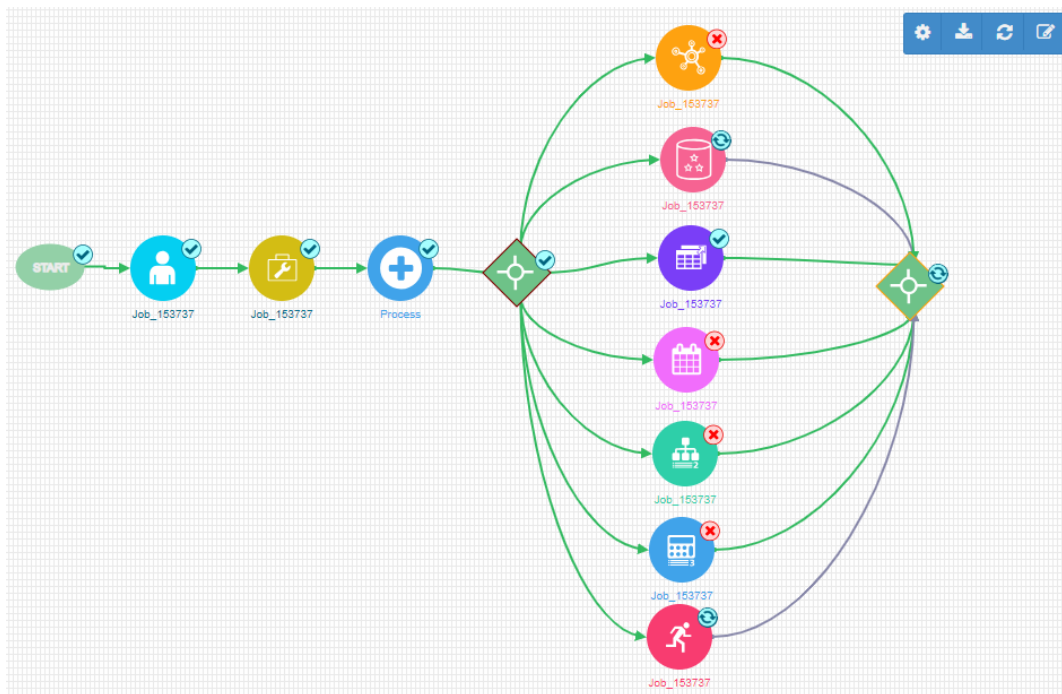
This window displays all the Workflows which are invoked from the Application with details such as Entity Name, Entity ID, Process Name, Process Description, Execution Start Time, Last Execution Time, Last Updated By and Status.

Using the *Search* grid, you can search for a Process/Workflow based on Process ID, Process Name, Application, Entity Name, Entity ID and Status.





You can sort the Processes displayed in the *Process Monitor* window based on Entity Id, Entity Name, Process ID or Process Name.



15.1 Monitoring a Business Process

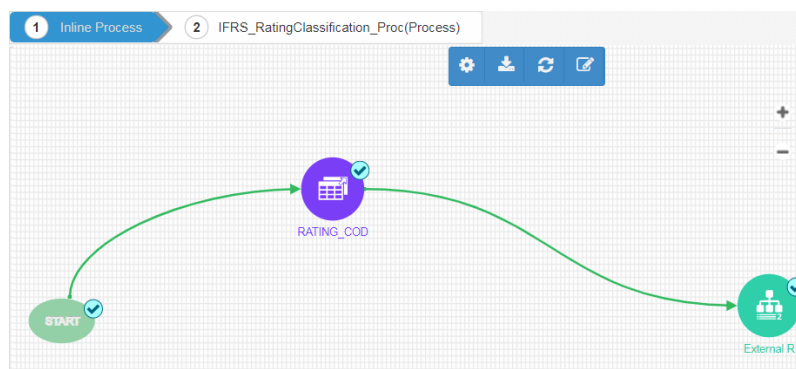
1. From the *Process Monitor* window, click the Entity ID link corresponding to the process you want to monitor.



In the *Process Monitor* window, status of activity is represented as given below:

-  - This indicates that the execution of activity is successful.
-  - This indicates that the activity is currently running or waiting for user's input to proceed.
-  - This indicates that the execution of activity is failed.
-  - This indicates that the activity is yet to be executed.

Double click Inline Process icon  or Sub Pipeline icon  to monitor the tasks inside them.





This figure shows the status of the tasks inside an Inline Process.

2. Click  to refresh the pane. Click  to view the *Process Logs* pane.

```
Workflow Instantiated..Jsontest_efaa12da-0e59-4f06-b8d7-16fd856ffc72
Starting Process Execution
[2018-04-09 08:40:42.112] *** Activity Execution Begins
Now...123456..Initiated By User..[AAAIUSER]
[2018-04-09 08:40:42.186] *** Activity execution
completed...123456 ***
[2018-04-09 08:40:42.209] *** Activity Execution Begins
Now...1234567..Initiated By User..[AAAIUSER]
All transitions from this activity are validated. No further
transitions...
[2018-04-09 08:40:42.27] *** Activity execution
completed...1234567 ***
```

The *Process Logs* pane shows all the execution stages of the process. Click the Process Monitor pane to close the *Process Logs* pane.

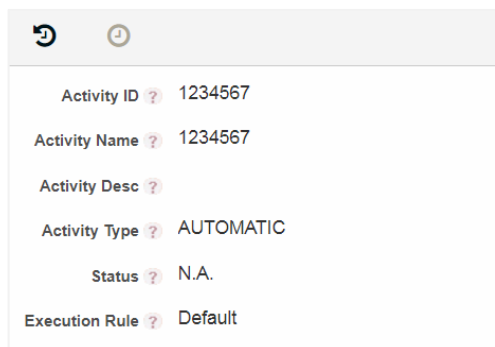
You can use  and  to zoom in and out the *Process Monitor* window.

15.2 Viewing Activity Logs

This feature allows you to view execution logs of each activity from the Process Monitor window.

To view Activity logs

1. Double-click on the activity icon whose logs you want to view. The Activity Definition details are displayed.



2. Click  to view the Activity Logs.

```
[2018-04-09 07:24:25.802] *** Activity Execution Begins
Now...RLBL6301..Initiated By User..[AAAIUSER]

[]

PreProcessing Activity Before Executing Task.
Preparing to Execute Pre Task for the Activity
Executing Starts:.....Job_1523249056394
Activity Has No Execution Task To Execute:.....
*****Preparing to Send ON ENTRY tasks:*****
No Task Available
Preparing to Execute Execution Rule for the Activity
Executing Starts:.....Job_1523249056394
Unable to get Valid Application Declaration: - Not Executing
Activity Rule
Started Executing OFSAA Activity: TYPE2
MIS Date: 04/09/2018
Execution class to be used:
com.ofs.aai.service.wf.ofsaa.component.RuleTypeTaskImpl
Initializing ICC Server for Execution..
Fetching task parameters for iccComponentId:
RULE_EXECUTION
Setting task parameters for iccComponentId: RULE_EXECUTION
- 1404823855822
Executing batch for ofsaa component: RULE_EXECUTION
RULE_EXECUTION Execution inProgress and the BatchId is
:OFSAAAINFO_1404823855822
```

The log shows all the execution stages of the selected Activity, if it is already executed. Otherwise, it will be blank.

16 Invoking PMF Pipeline

16.1 Application UI

Application can invoke PMF in the following ways:

1. Java API

```
WorkflowEngineAPI.startWorkflowProcess( String objectId, String
objectType , String infodom, String segment, String userID, String
locale,Map<String, String> applicationParams,Map securityMap)
```

2. Stored Procedure

```
create or replace procedure startWorkflowProcessAsynch(objectId IN VARCHAR2,
objectType IN VARCHAR2,
infodom IN VARCHAR2,
segments IN VARCHAR2,
userID IN VARCHAR2,
locale IN VARCHAR2,
appParams IN array_varchar,
secMap IN array_varchar) is
```

3. Rest Service

URL: <contextPath>/PMFService/startWorkflowProcess

Method: POST

Consumes ("application/json")

Produces ("text/plain")


Sample Input Params:

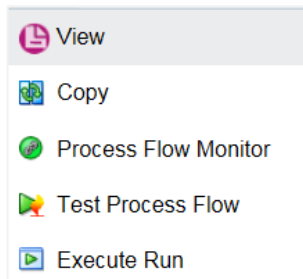
```
"{"SummaryPayload":{"objectId":"123","objecttype":"QTNR",
"infodom":"OFSAAIINFO","segment":"OFSAAIINFO","userid":
"AAAIUSER","locale":"en_US","securitymap"
: {}, "applicationparams"
: {"testparam":"value1","testparam2":"value2"}}}";
```

16.2 Within PMF Summary Screen UI

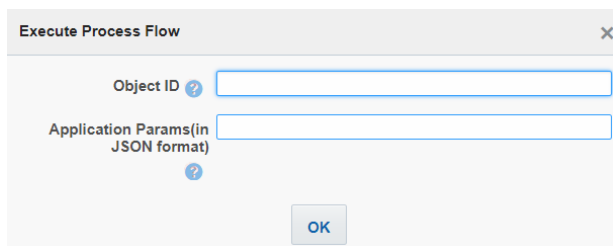
16.2.1 Using Test Workflow

Testing Process Flow option is used to check whether the Process flow you designed works as expected.

1. From the *Process Modeller* window, click  corresponding to the Pipeline you want to test. The Sub menu is displayed.



2. Click **Test Process Flow**. The *Execute Process Flow* window is displayed.



3. Enter an **Object ID** to identify the Process flow. This will be displayed as Entity ID in the Process Monitor window.
4. Enter the **Application Parameters** which are required in the Process flow that you are testing. Specify parameters in JSON format. This is an optional field.

For example, {"WF_RUNSK": "15", "WF_MISDATE": "12/31/1999"}

5. Click **OK**.

16.2.2 Using Execute Run

See [Executing Run Pipeline](#) section on how to invoke a Run pipeline from the PMF screen.

16.3 Operations Module

Execution of a PMF Pipeline can be triggered from the Operations module as a batch.

1. From the *Batch Maintenance* window under Operations module, create a new Batch. For more information, see *Adding Batch Definition* section in the [OFS Analytical Applications Infrastructure User Guide](#).

2. Create a new Task with task component as Workflow Execution. For more information, see [Adding Task Details](#) section in the [OFS Analytical Applications Infrastructure User Guide](#).

Task Definition

Save Reset Close

Task ID: Task1 Description: PMF Task

Components: WORKFLOW_EXECUTION

Dynamic Parameters List

Property	Value
Datastore Type	EDW
Datastore Name	OFSIFRSINFO
Primary IP For Runtime Processes	whf00aix
Object ID	
Workflow	pmf_cecl_run
Optional Parameters	

Audit Panel

Created By: Creation Date

Last modified by: Last Modification Date

- a) Select the PMF Pipeline you want to execute from the **Workflow** drop-down list.
 - b) Enter any parameters you want to pass during the execution of the Pipeline in the **Optional Parameters** field.
 - c) Click **Save**.
3. Execute the Batch from the *Batch Execution* window. For more information, see *Batch Execution* section in the [OFS Analytical Applications Infrastructure User Guide](#).

16.4 Command Line Execution

A shell script file `wfExec.sh` is available in `ficdb/bin` folder.

To execute the utility, navigate to `$FIC_DB_HOME/bin` and execute `wfExec.sh` with parameters such as `objectId`, `objectType`, `Infodom`, `segments`, `userID`, `locale`, `appParams`, and `secMap`.

17 Appendix A

17.1 Enabling Proxy for REST Service Application Rule

This section explains how to configure the Proxy details if it is required for the Rest Service Application Rule.

Add the following entries in the `AAI_WF_GLOBAL_SETTINGS` table:

V_PARAM_NAME	V_PARAM_VALUE	Description
PROXY_SERVER_IP	For example, www.proxy.myserver.com	Provide the IP address of the Proxy server.
PROXY_SERVER_PORT	For example, 80	Provide the port number of the Proxy server.


17.2 Configuring Group Approval for Human Tasks

Group approval/Group Consensus can be used to decide whether the flow has to move to a particular activity based on the response of a single member, majority of members of the group, or all members of the group. If more than one group is present, then you can design to move the flow to a particular activity based on the response of a single group, all groups or the majority of groups.

17.2.1 Configuring Parallel Group Approval

Parallel group approval is used when you want to send the task to all users in the task group simultaneously.

To configure parallel group approval:

1. From the *Process Modeller* window, select the required Process and click **Edit**. The *Process Flow* tab is displayed.
2. Double-click the Activity for which you want to configure user approval.
3. On the right pane, click . The *Actions* tab is displayed.
4. Click **Group Approval** to configure group approval. The *Task Stage Details* window is displayed.

Task Stage Details

Routing Policy

? Parallel Sequential

Parallel Voting Policy

+ Add | - Delete

Select	Target Activity	Voting Formula	Value
<input type="radio"/>	audit approval	Approved by anyone	

< Default Target Activity ? >

Sequential Voting Policy

Ok

4. Select **Parallel** to configure parallel group approval.
5. Click **Add**. A row is added to define the voting formula for target activity.

NOTE: It is recommended that you define voting formula for all activities. If voting formula is not defined for an activity and if someone in a task group selects that activity, the workflow moves to that activity.


6. Select the required **Target Activity** from the drop-down list.
7. Select the required option from the **Voting Formula** drop-down list. The options are:
 - **Approved by anyone-** If any one of the users from any task group chooses the selected **Target Activity**, the flow moves to the selected activity. If no one chose it, it checks for the voting formula defined for the next Target Activity.
 - **Approved by anyone from every Group-** If at least one user from every task group chooses the selected **Target Activity**, the flow moves to that activity.
 - **Approved by overall majority-** If majority of the users from all task groups choose the selected **Target Activity**, the flow moves to that activity. For example, if there are 2 task groups and 15 users in each group, then at least 16 users (majority of 30 users) should choose the selected **Target Activity**, for the flow to proceed to that activity.
 - **Approved by majority from each Group-** If majority of the users from each task group choose the selected **Target Activity**, the flow moves to that activity. For example, consider there are 3 task groups and each group has 15 users, then from each group at least 8 users should favour the Target Activity to move the flow to that activity.
 - **Approved by everyone-** All the users in all the task group should choose the selected **Target Activity** for the flow to move to that activity.

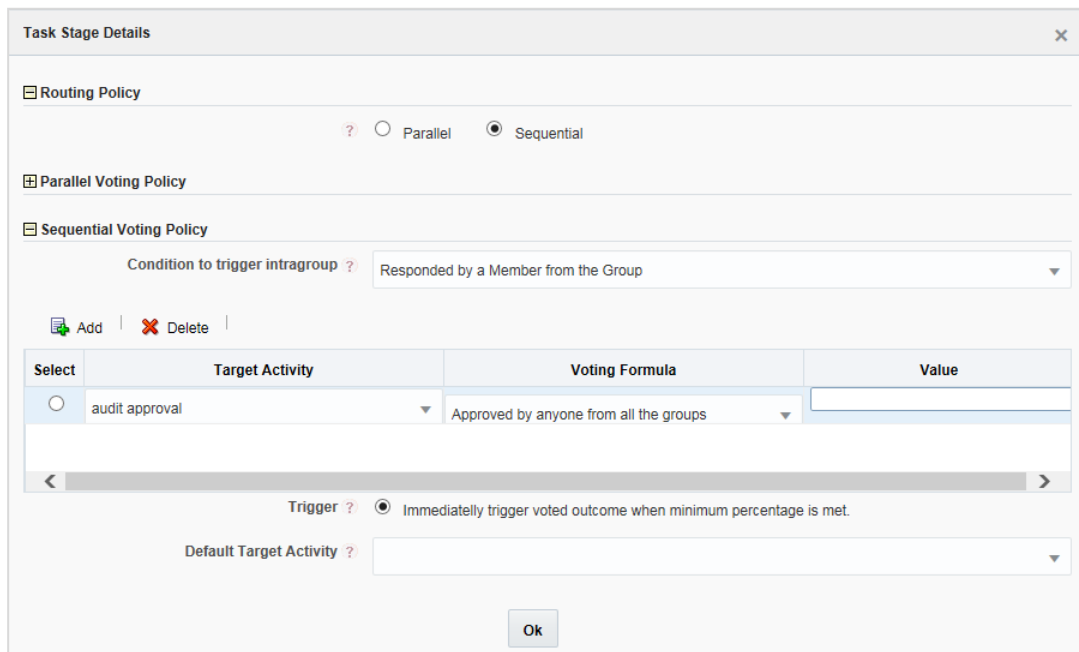
- **Approved by overall percentage-** If the specified percentage of users in the task group chooses the selected **Target Activity**, the flow moves to that activity. Enter the percentage in the **Value** field.
8. Select the **Default Target Activity** from the drop-down list. This is the activity which will be executed if none of the condition is satisfied.
 9. Click **Ok**.

17.2.2 Configuring Sequential Group Approval

Sequential Group approval is used when you have multiple tasks for an activity.

To configure sequential group approval:

1. From the *Process Modeller* window, select the required Process and click **Edit**. The *Process Flow* tab is displayed.
2. Double-click the Activity for which you want to configure user approval.
3. On the right pane, click . The *Actions* tab is displayed.
4. Click **Group Approval** to configure group approval. The *Task Stage Details* window is displayed.



The screenshot shows the 'Task Stage Details' window with the following configuration:

- Routing Policy:** Parallel, Sequential
- Parallel Voting Policy:** (Collapsed)
- Sequential Voting Policy:**
 - Condition to trigger intragroup: Responded by a Member from the Group
 - Buttons: Add, Delete
 - Table:

Select	Target Activity	Voting Formula	Value
<input type="radio"/>	audit approval	Approved by anyone from all the groups	
 - Trigger: Immediately trigger voted outcome when minimum percentage is met.
 - Default Target Activity: (Empty dropdown)
 - Button: Ok

5. Select **Sequential** to configure sequential group approval.
6. Select the **Condition to trigger intragroup** from the drop-down list. Intragroup is the task group of each task in a particular activity. The sequence in which each intragroup will be considered for voting is based on the sequence in which the tasks are added in the *Task Stage* pane in the *Actions* tab. The options are:

- **Responded by a Member from the Group-** If anyone from the task group of the first task responds, it goes to the next task group and wait till someone from that task group respond and so on.
- **Responded by Overall Majority-** If majority of users in the task group of the first task respond, it goes to the next task group and wait till majority of users in the task group of the second task respond and so on.
- **Responded by Everyone in each Group-** Once all users in the task group of the first task have responded, it moves to the next task group. Then it waits till everyone in the second task group respond and so on.

10. Click **Add**. A row is added to define the voting formula for target activity.

NOTE: It is recommended that you define voting formula for all activities. If voting formula is not defined for an activity and if someone in a task group selects that activity, the workflow moves to that activity.

11. Select the required **Default Target Activity** from the drop-down list. This is the activity which will be executed if none of the condition is satisfied.

12. Select the required option from the **Voting Formula** drop-down list. The options are:

- **Approved by anyone from all the groups-** If anyone selects the Target Activity, the flow moves to that activity.
- **Approved by Overall Majority-** If majority of the users select the Target Activity, the flow moves to that activity.
- **Approved by Everyone in each group-** If everyone in the group selects the Target Activity, the flow moves to that activity.
- **Approved by Overall Percentage-** Provide the percentage in the Value field. If the specified percentage of the users select the Target Activity, the flow moves to that activity.

13. Select the activity which needs to be executed if the condition fails, from the **Default Target Activity** drop-down list.

17.3 JsonPath Expressions

JsonPath expressions always refer to a JSON structure in the same way as XPath expression are used in combination with an XML document. The "root member object" in JsonPath is always referred to as \$ regardless if it is an object or array.

JsonPath expressions can use the dot-notation

```
$.store.book[0].title
```

or the bracket-notation

```
$['store']['book'][0]['title']
```

17.3.1 Operators

Operator	Description
\$	The root element to query. This starts all path expressions.
@	The current node being processed by a filter predicate.
*	Wildcard. Available anywhere a name or numeric are required.
..	Deep scan. Available anywhere a name is required.
.<name>	Dot-notated child
['<name>' (, '<name>')]	Bracket-notated child or children
[<number> (, <number>)]	Array index or indexes
[start:end]	Array slice operator
[?(<expression>)]	Filter expression. Expression must evaluate to a boolean value.

17.3.2 Functions

Functions can be invoked at the tail end of a path - the input to a function is the output of the path expression. The function output is dictated by the function itself.

Function	Description	Output
min()	Provides the min value of an array of numbers	Double
max()	Provides the max value of an array of numbers	Double
avg()	Provides the average value of an array of numbers	Double
stddev()	Provides the standard deviation value of an array of numbers	Double
length()	Provides the length of an array Integer	Integer

17.3.3 Filter Operators

Filters are logical expressions used to filter arrays. A typical filter would be [?(@.age > 18)] where @ represents the current item being processed. More complex filters can be created with logical operators && and ||. String literals must be enclosed by single or double quotes


```
([?(@.color == 'blue')] or [?(@.color == "blue")])
```

Operator	Description
==	left is equal to right (note that 1 is not equal to '1')
!=	left is not equal to right
<	left is less than right
<=	left is less or equal to right
>	left is greater than right
>=	left is greater than or equal to right
==~	left matches regular expression [?(@.name =~ /foo.*?/i)]
in	left exists in right [?(@.size in ['S', 'M'])]
nin	left does not exists in right
subsetof	left is a subset of right [?(@.sizes subsetof ['S', 'M', 'L'])]
size	size of left (array or string) should match right
empty	left (array or string) should be empty

17.3.4 Path Examples

```
{
  "store": {
    "book": [
      {
        "category": "reference",
        "author": "Nigel Rees",
        "title": "Sayings of the Century",
        "price": 8.95
      },
      {
        "category": "fiction",
        "author": "Evelyn Waugh",
```

```
        "title": "Sword of Honour",
        "price": 12.99
    },
    {
        "category": "fiction",
        "author": "Herman Melville",
        "title": "Moby Dick",
        "isbn": "0-553-21311-3",
        "price": 8.99
    },
    {
        "category": "fiction",
        "author": "J. R. R. Tolkien",
        "title": "The Lord of the Rings",
        "isbn": "0-395-19395-8",
        "price": 22.99
    }
],
    "bicycle": {
        "color": "red",
        "price": 19.95
    }
},
    "expensive": 10
}
```

17.4 Delegation

This feature facilitates you to delegate the tasks/notifications assigned to you to another user. The delegate can be your peer, someone from your immediate subordinate or someone from your all subordinates. Additionally, you can revoke active delegations whenever required. Your user group should be mapped to the function role WFDELACC (Process Delegation User) if you want to define delegation.

Financial Services Analytical Applications Infrastructure > Processing Modelling Framework > Delegation

Add |
 View |
 Edit |
 Delete |
 Revoke

Selk	Delegator	Delegator	Delegate To	Start Date	End Date	Status
<input type="radio"/>	1475137557	ORMUSER	CP	2016-10-01	2016-10-14	REVOKED
<input type="radio"/>	1475146608	ORMUSER	undefined	2016-09-29	2016-09-29	ACTIVE

This window displays all the delegations which are defined by the logged in user with details such as Delegation ID, Delegator, Delegate To, Start Date, End Date, and Status. You can add a new Delegation, view, modify, delete and revoke a delegation.

17.4.1 Adding a Delegate

To add a delegate

1. From the *Delegation* window, click **Add**. The *Delegation Details* window is displayed.

Delegation Details ✕

Delegation ID ? 1477473481896

Delegator ? Tom Harley

Identify Delegate ?

Delegate To ?

Start Date ? 10/26/16

End Date ? 10/26/16

Notification Required ?

Filter ?

Notification Message ?

Scope ?

Application ?

Process ? All

Comments ?

2. Enter the details as tabulated:

Field Name	Description
Delegation ID	Displays the auto generated Delegation ID.
Delegator	Displays the User ID of the logged-in user. If your user group is mapped to the function role WFDELGADM, you can select the delegator from the drop-down list.
Identify Delegate	Select the required option from the drop-down list. The options are: <ul style="list-style-type: none"> ▪ Peers – Select this option if you want to delegate your tasks to your peer, who reports to your own manager. ▪ Subordinates- Select this option if you want to delegate your tasks to your immediate subordinates. ▪ All subordinates – Select this option if you want to delegate your tasks to someone who comes under you in your organization.
Delegate To	Select the user to whom you want to delegate your tasks from the drop-down list. Based on the selected option from the Identify Delegate drop-down list, the users are displayed in this drop-down list. For example, if Peers is selected as Identify Delegate, this drop-drown list displays all the peers in your organization. The data is fetched from <code>AAI_EMPLOYEE_MASTER</code> table.
Start Date and End Date	Specify the duration for which you want to delegate your tasks by selecting the Start Date and End Date from the calendar.
Notification Required	Turn ON the toggle button if you want to send a notification to the delegate or delegator.
Filter	This field is enabled only if the Notification Required toggle button is turned ON. Select to whom you want to send the notification. You can set to send notification to Delegator and Delegate . Select when you want to send the notification. The options are On Defining and On Execution .
Notification Message	This field is enabled only if the Notification Required toggle button is turned ON. Select the notification message you want to send to the delegate or delegator.

Field Name	Description
Scope	<p>Select the scope of the delegation from the drop-down lists. The options are:</p> <ul style="list-style-type: none"> ▪ All- Select this option to delegate all your tasks. ▪ Application- Select this option if you want to delegate all your tasks for a particular Application only. ▪ Process- Select this option if you want to delegate all your tasks for a particular Process only.
Application	<p>This field is enabled only if Application or Process is selected as Scope.</p> <p>Select the required Application from the drop-down list. All your tasks related to the selected application will be delegated to the selected user.</p>
Process	<p>This field is enabled only if Process is selected as Scope.</p> <p>Select the required Process from the drop-down list. The list displays all processes related to the selected Application. All your tasks related to the selected process will be delegated to the selected user.</p>
Comments	<p>Enter if you want to add any comments for the delegation.</p>

3. Click **Save**.


17.4.2 Viewing Delegation

This option allows you to view the details of existing delegations.

From the *Delegation* window, select the required delegation and click **View**. You can view the Delegation details.

17.4.3 Modifying Delegate Details


To modify delegate details

1. From the *Delegation* window, select the delegation you want to modify and click  **Edit**. The Delegation Details window is displayed.
2. Modify the required details. For more information, see [Adding a Delegate](#) section.

17.4.4 Revoking Delegation

You can revoke only active delegations.


To revoke delegation

1. From the *Delegation* window, select the delegation you want to revoke and click  **Revoke**.
2. Click **OK** in the confirmation message box.

17.4.5 Deleting Delegation

You cannot delete active delegations.

To delete a delegation

1. From the *Delegation* window, select the delegation you want to delete and click  **Delete**.
2. Click **OK** in the confirmation message box.



OFSAAI
Process Modelling Framework Orchestration Guide

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com/financial_services/

Copyright © 2018 Oracle Financial Services Software Limited. All rights reserved.

No part of this work may be reproduced, stored in a retrieval system, adopted or transmitted in any form or by any means, electronic, mechanical, photographic, graphic, optic recording or otherwise, translated in any language or computer language, without the prior written permission of Oracle Financial Services Software Limited.

Due care has been taken to make this OFSAAI Process Modelling Framework Orchestration Guide and accompanying software package as accurate as possible. However, Oracle Financial Services Software Limited makes no representation or warranties with respect to the contents hereof and shall not be responsible for any loss or damage caused to the user by the direct or indirect use of this OFSAAI Process Modelling Framework Orchestration Guide and the accompanying Software System. Furthermore, Oracle Financial Services Software Limited reserves the right to alter, modify or otherwise change in any manner the content hereof, without obligation of Oracle Financial Services Software Limited to notify any person of such revision or changes.

All company and product names are trademarks of the respective companies with which they are associated.
