Oracle® Retail Process Orchestration and Monitoring

Implementation Guide

Release 19.1

F35029-01

September 2020

ORACLE®

Oracle® Retail Process Orchestration and Monitoring Implementation Guide, Release 19.1

F35029-01

Primary Author:

Contributing Author:

Contributor:

Applications. For purposes of this section, "alteration" refers to all alterations, translations, upgrades, enhancements, customizations or modifications of all or any portion of the VAR Applications including all reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

The VAR Applications contain trade secrets of Oracle and Oracle's licensors and Customer shall not attempt, cause, or permit the alteration, decompilation, reverse engineering, disassembly or other reduction of the VAR Applications to a human perceivable form. Oracle reserves the right to replace, with functional equivalent software, any of the VAR Applications in future releases of the applicable program.

Contents

Send Us Your Comments

Oracle® Retail Process Orchestration and Monitoring Implementation Guide, Release 19.1

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document.

Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

> **Note:** Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the Online Documentation available on the Oracle Technology Network Web site. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at http://www.oracle.com.

Preface

This Implementation Guide describes the requirements and procedures to install this Oracle Retail Product release.

Audience

This Implementation Guide is for the following audiences:

- System administrators and operations personnel
- Database administrators
- System analysts and programmers
- Integrators and implementation staff personnel

Documentation Accessibility

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

Access to Oracle Support

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

Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

https://support.oracle.com

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you take

Improved Process for Oracle Retail Documentation Corrections

To more quickly address critical corrections to Oracle Retail documentation content, Oracle Retail documentation may be republished whenever a critical correction is needed. For critical corrections, the republication of an Oracle Retail document may at times not be attached to a numbered software release; instead, the Oracle Retail document will simply be replaced on the Oracle Technology Network Web site, or, in the case of Data Models, to the applicable My Oracle Support Documentation container where they reside.

This process will prevent delays in making critical corrections available to customers. For the customer, it means that before you begin installation, you must verify that you have the most recent version of the Oracle Retail documentation set. Oracle Retail documentation is available on the Oracle Technology Network at the following URL:

http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html

An updated version of the applicable Oracle Retail document is indicated by Oracle part number, as well as print date (month and year). An updated version uses the same part number, with a higher-numbered suffix. For example, part number E123456-02 is an updated version of a document with part number E123456-01.

If a more recent version of a document is available, that version supersedes all previous versions.

Oracle Retail Documentation on the Oracle Help Center (docs.oracle.com)

Oracle Retail product documentation is also available on the following Web site:

https://docs.oracle.com/en/industries/retail/index.html

(Data Model documents can be obtained through My Oracle Support.)

Conventions

The following text conventions are used in this document:

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

1
Introduction

The Process Orchestration and Monitoring (POM) application provides comprehensive batch scheduling and monitoring capabilities for Oracle Retail SaaS Applications.

POM's key features are:

- Support for Cyclical (or Hourly)/Ad-hoc (or Standalone)/End of Day (or Nightly) scheduling modes
- Configurable Schedule Invocation - Time/Externally triggered
- Configurable Schedule - Configure schedule specific to customer needs
- Hooks to integrate with external (customer) systems - for dependency management and job status updates
- Configurable notifications

2

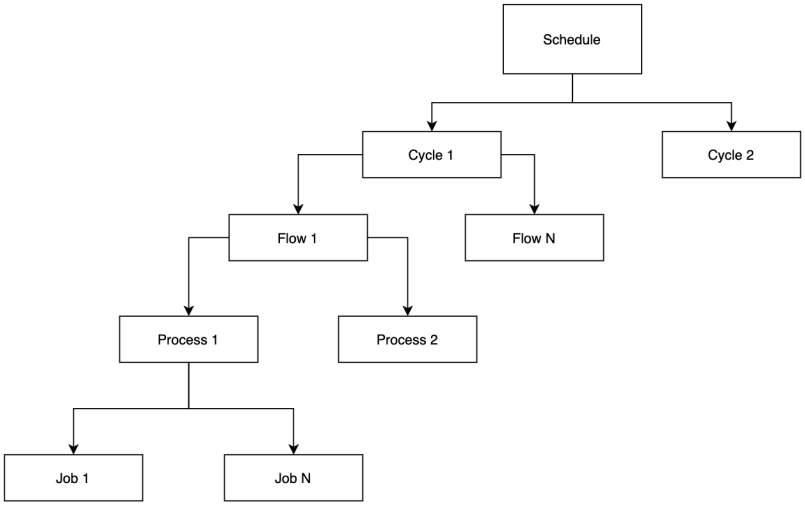
Batch Concepts

This chapter describes the Process Orchestration and Monitoring (POM) concepts that are key to configure and implement the product successfully.

Batch Hierarchy

POM logically groups batch jobs in a schedule into a hierarchy of Processes, Flows and Cycles based on functionality and expected execution frequency.

Figure 2–1 Batch Hierarchy Diagram



Cycle

A Batch Cycle is a logical collection of flows. Cycles are categorized into 3 groups,

1. **Ad hoc/Standalone**

   The jobs that can run multiple times a day on an as-needed basis are classified as Ad hoc cycles. Such processes are allowed to run independent of other processes in a Flow/Cycle

2. **Recurring/Hourly**

   Recurring or Hourly jobs are a special group of jobs which run multiple times a day at scheduled intervals.

Example: The sales processing jobs in RMS support trickle processing by running every 30 minutes during the store trading hours. The schedule is pre-loaded with 24 cycles. Each cycle can have one or more flows where each flow is a set of jobs. The recurring cycles are time triggered from the Scheduler. Based on the client's business operations, the individual flows of each cycle will need to be scheduled. If any flow/cycle is not used then it has to be disabled as the nightly cycle will not start until the enabled cycles are all complete.

3. **End of Day/Nightly**

   This contains the set of jobs that are executed at end of the business day. The nightly cycle is time triggered from Scheduler. The nightly cycle will start only when all the loaded recurring flows are complete.

Flow

A batch flow is a logical collection of batch processes that run together. The batch flow always starts and ends with a single process, but it can run parallel processes in the middle of the batch flow.

Process

A batch process is a logical collection of batch jobs that always run in sequential order.

Job

A batch job is smallest entity in POM that can be scheduled. A job represents an actual application script.

Internal Dependencies

Internal dependencies are the dependencies between the process/job of the same schedule.

Inter-Schedule Dependencies

Inter-Schedule dependencies are the dependencies between jobs of the different schedules running on the same POM instance.

External Dependencies

External dependencies are the dependencies between the jobs running on external systems, such as a customer's system and jobs running on the POM schedule.

3

Integration

This chapter describes the various scenarios which involve configuring & integrating the Process Orchestration and Monitoring (POM) application to connect with external systems.

Invoking Cycles in POM

Generally, the Oracle support team configures the time and frequency for running the batch cycles in POM on behalf of customers based on their requirement.

However, the capability exists in POM for customers if they so desire to control the time and frequency of batch executions by invoking the following ReST service.

Execution Request Creation

The specification of the ReST service to start the POM cycle execution is shown below:

HTTP Method	POST
Path	http://**<pom-server-host>**/ProcessServices/services/private/executionEngine/schedules/**<Schedule_Name>**/execution **Note: <pom-server-host>** and **<Schedule_Name>** variables need to be changed according to the target environment and schedule.
HTTP Headers	Content-Type = application/json Accept-Version = 16.0

Request Body	`{`
	`"cycleName": "<Cycle Name>",`
	`"flowName": "<Flow Name>",`
	`"processName": "<Process Name>",`
	`"requestParameters":"<Comma Separated Key Value Pairs>"`
	`}`
	
	Notes:
	▪ **processName** attribute required only for Ad hoc cycles
	▪ **requestParameters** attribute is optional. This is useful if external systems would like to provide custom identifiers to POM and expect them to be returned on callbacks sent from POM.

Response Body	{
	"value": "171" **-Execution Request ID,**
	"cycleName": "CycleName",
	"flowName": "FlowName",
	"processName": "ProcessName",
	"jobName": "JobName",
	"requestType": "requestType",
	"requestParameters": ":"<Comma Separated Key Value Pairs>"
	"executionEngineInfo": "STARTED",
	"hyperMediaContent": {}
	}

Invocation	Request Payload

Nightly Cycle invocation	{
	"cycleName" : "Nightly",
	"flowName" : "Nightly",
	"requestParameters" : "callerId=XXX,correlationId=123"
	}
	
	Note: The Nightly Cycle contains a single flow hence a single invocation will suffice to start the Nightly Flow.

Hourly Cycle invocation	{
	"cycleName" : "Hourly_Cycle_**<N>**",
	"flowName" : "**<FLOW_NAME>**",
	"requestParameters" : "callerId=XXX,correlationId=456"
	}
	
	<N> is the cycle number (1 to 24)
	<FLOW_NAME> is name of the flow (Process Group) to invoke. For example, SALESPROCESS_CYCLE.
	{
	"cycleName" : "Hourly_Cycle_1",
	"flowName" : "SALESPROCESS_CYCLE",
	"requestParameters":"callerId=Sys_Name,correlationId=456"
	}
	
	Note: The Hourly Cycles comprise of many distinct flows and for each flow a separate invocation would be required.

Invocation	Request Payload

Ad hoc Cycle invocation	{
	"cycleName" : "Adhoc",
	"flowName" : "Adhoc",
	"processName" : "<ADHOC_PROCESS_NAME>",
	"requestParameters" : "callerId=xxx,correlationId=789"
	}
	
	Example:
	{
	"cycleName": "Adhoc",
	"flowName":"Adhoc",
	"processName":"RPM_NEW_ITEM_LOCATION_PROCESS _ADHOC",
	"requestParameters":"callerId=Ext_System_Name,
	correlationId= 789"
	}
	
	Note: Ad hoc Cycles are composed of many discrete individual Processes. For each Process, a separate invocation is required.

Execution Request Status Tracking

The following endpoints provide the ability to check the status of an execution request in POM:

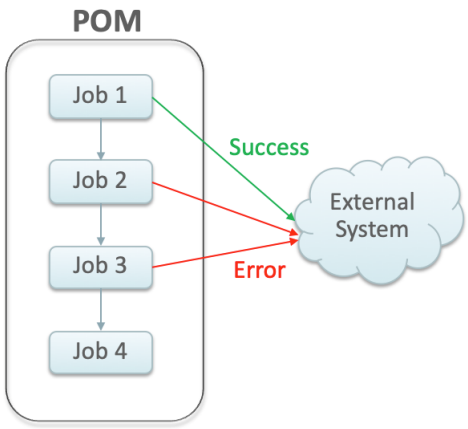
HTTP Method	GET
Path	http://**<pom-server-host>**/ProcessServices/services/private/executionEngine/schedules/**<scheduleName>**/requests/{executionId} **Note:** Replace **<pom-server-host>** according to the target environment. **<scheduleName>**: Name of the schedule. For MOM apps, the schedule name is "MERCH". For RDE, the schedule name is "RDE". **<executionId>**: ID of the execution request returned by POM when execution request was submitted.
HTTP Headers	Content-Type = application/json Accept-Version = 16.0

Response Body	■ **executionId** - ID of the execution request ■ **scheduleName** - Name of the schedule for which this execution request was created. **Sample values:** MERCH, RDE, and so on. ■ **cycleName** - Name of the Cycle for which this execution request was created. **Sample values:** Nightly, Hourly, or Adhoc. ■ **flowName** - Name of the Flow for which the execution request was created **Sample Values:** For Ad hoc Cycle: Adhoc For Nightly Cycle: Nightly For Hourly Cycle: Name of the flow such as DEAL_PURGE_CYCLE, and so on. ■ **processName** - Name of the Process for which the execution request was created. **Sample Values:** For Ad hoc: Name of the process such as POINDBATCH_PROCESS_ADHOC, and so on. For Nightly/Hourly: Always set to "ALL". ■ **requestParameters** - Parameters associated with the execution request. ■ **status** - Status of the execution request. **Possible Values:** **QUEUED**: Request is queued up for execution. **RUNNING**: Jobs from this request are being executed **ERROR**: One of the job in this request has failed. Note that a failed job would be restarted by POM Admin; there is no need to re-submit the execution request. **COMPLETED**: All jobs from this request were executed successfully.

External Status Update

The External Status Update feature provides the ability for external systems to register with POM to receive the Job status notifications as a callback to their ReST interface.

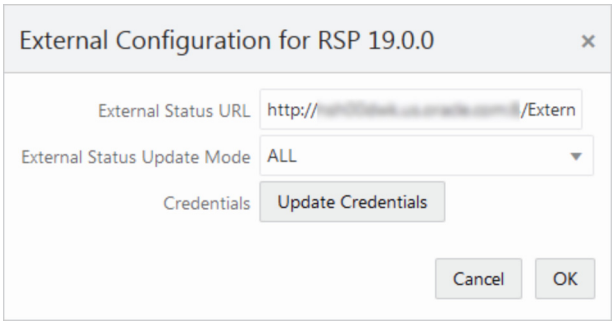
Figure 3–1 External Dependency



Schedule Configuration

This section details the steps to configure the External Status Updates feature at the schedule level:

1. Navigate to the System Configuration screen.
2. Click the Edit icon on the External Configuration Panel to open the External Configuration window.



3. Enter the configuration values:
   - **External Status URL** - External system's URL that needs to be called for status updates.

     Note: In addition to this configuration, you must work with Oracle support to get the External Status URL allowlist.

   - **External Status Update Mode** - Choose one of the options below:
     - **ALL** - POM will send a status update to the external system for each job's execution in the schedule regardless of success or failure.
     - **FAILED** - POM will notify only for failed jobs.
     - **NONE** - No status updates will be sent by POM.

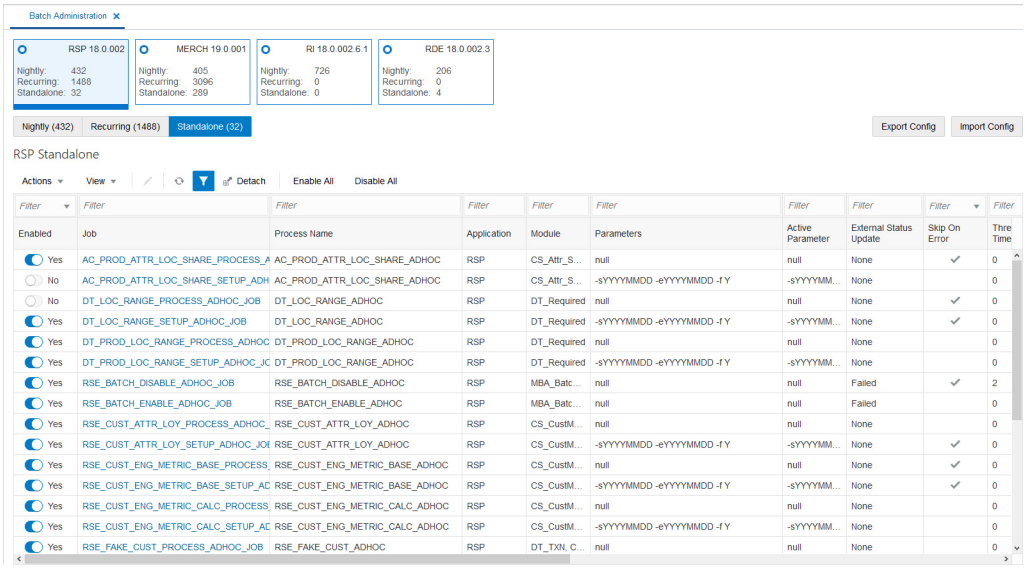
Note: The External Status Update Mode defined on this screen applies to all the jobs in a schedule. If status update is desired only for specific jobs then set the mode on the above screen to NONE and follow the steps defined in the Job Configuration section below.

- Click **Update Credentials** and provide the credentials for the external system.

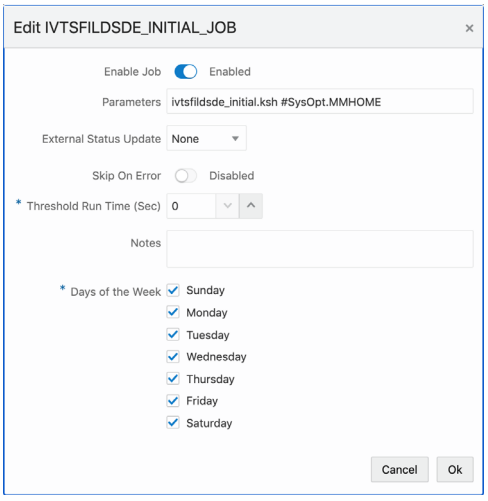
Job Configuration

This section describes the steps to configure the External Status Update Mode at job level.

1. Navigate to the Batch Administration screen and select the schedule to update with the external status.



2. Select one of the Cycle - Nightly/Recurring/Standalone
3. Select the **Process/Job** combination and click Edit from table action menu to open the popup below.



4. Set the **External Status Update Mode** one of the following values:
   - **ALL** - POM will send a status update to the external system for each job's execution regardless of success or failure.
   - **FAILED** - POM will notify only for when this job fails.
   - **NONE** - No status update will be sent by POM for this job.

Payload Specification

Attribute	Description
processName	Name of the root process in a given cycle/flow **Note:** Process names in the callback response are prefixed with the name of the schedule. For instance, a callback response sent for Process "P1" would have processName attribute as MERCH_P1 on return.
processExecutionId	Unique identifier generated by POM to track the process executions.
activityName	Name of the job for which the callback/status update is sent.
activityExecutionId	Unique identifier generated by POM to track the job run instance.
callerId	Identifier provided by the caller to POM when submitting the invocation/execution request. POM returns the same ID back to caller.
correlationId	Identifier provided by the caller to POM when submitting the invocation/execution request. POM returns the same ID back to caller
callBackServiceDataDetail.<KeyName>	Key value pairs supplied to POM when submitting the invocation/execution request. Those are returned back to the caller
failedActivity	In the case where the callback is for a failed job, this field is populated with the details of the failed Job.
status	Status of the job execution - COMPLETED"SKIPPED, ERROR SKIPPED_ON_ERROR
activityStatus	Status of the job, and the derived activity state: ■ ERROR - ACTIVITY_FAILED ■ COMPLETED - ACTIVITY_COMPLETED ■ SKIPPED - ACTIVITY_COMPLETED ■ SKIPPED_ON_ERROR - ACTIVITY_COMPLETED

Payload Examples

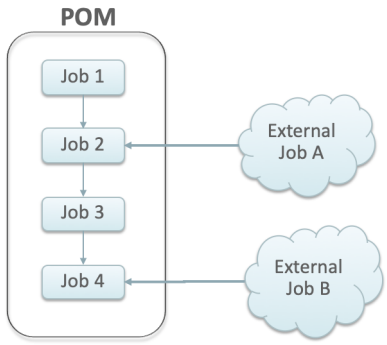
Below are sample external status update payloads for the MERCH schedule.

Description	Payload
Hourly Job Callback	{ " callerId " : "XXX", " correlationId " : "37", " processName " : "MERCH_ BATCH_PROCESS _01", " processExecutionId " : "MERCH_ BATCH_PROCESS_01 ~asfasfdasdfas", " activityName " : " MERCH_ BATCH_JOB", " activityExecutionId " : "123456", " status " : "COMPLETED", " activityStatus " : "ACTIVITY_COMPLETED", " failedActivity " : null }
Nightly Job Callback	{ " callerId " : "XXX", " correlationId " : "37", " processName " : "MERCH_ START_NIGHT_BATCH_PROCESS ", " processExecutionId " : "MERCH_ START_NIGHT_BATCH_ PROCESS ~asfasfdasdfas", " activityName " : " MERCH_ START_NIGHT_BATCH_JOB", " activityExecutionId " : "123456", " status " : "COMPLETED", " activityStatus " : "ACTIVITY_COMPLETED", " failedActivity " : null }

External Dependency

This feature allows customers to control the execution of a schedule running in POM by defining custom pre-dependencies. POM pauses the schedule execution on encountering these external pre-dependences and resumes the execution once they are released by customer.

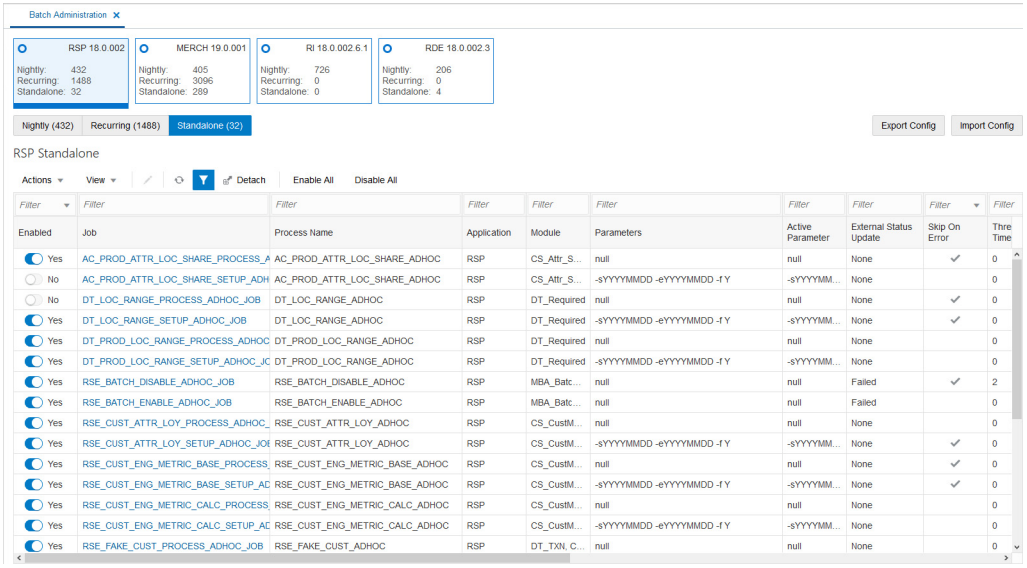
Figure 3–2 External Dependency



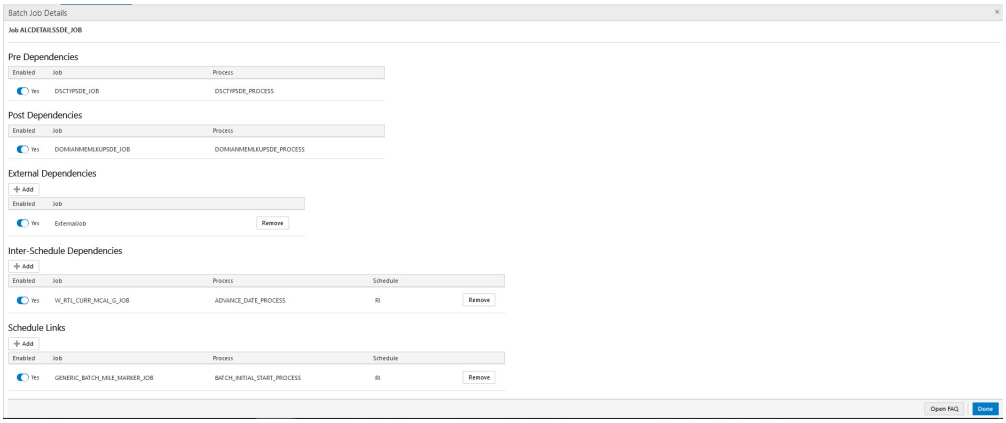
Configuration

This section details the steps involved in setting up the external dependency.

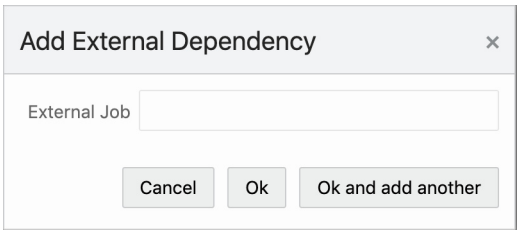
1. Navigate to the Batch Administration screen and select the schedule to which the external dependency will be added.



2. Select a Cycle - Nightly/Recurring/Standalone.
3. Find the **Process/Job** combination to which dependency needs to be added.
4. Click on the **Job** name to open the Batch Job Details panel.



5. On the External Dependency section of the screen, click the **Add** button to create the external dependency.
6. Provide the external job name.



7. Click **Ok** to save and exit or **Ok and add another** to create another dependency.

Releasing Dependency

External systems need to invoke the following POM ReST service to release/fulfill the external dependencies.

HTTP Method	POST
URL	http://**<pom-server-host>**/ProcessServices/services/private/schedules/**<Schedule_Name>**/external/jobs/**<Ext_Dependendy_Name>**/status/COMPLETED **Note: <pom-server-host>**, **<Schedule_Name>** and **<Ext_Dependendy_Name>** variables need to be changed according to the target environment and schedule.
HTTP Headers	Content-Type = application/json Accept-Version = 16.0
Request Body	None

Response Body	<pre>{
	"value": "true",
	"links": [],
	"hyperMediaContent" : { "linkRDO": []}
	}</pre>
	The true/false in the value attribute shown above indicates the success/failure of releasing the external dependency respectively.

4

Schedule Customization

All applications that run batch cycles in POM maintain a standard batch schedule with full list of jobs, pre-defined dependencies, job parameters, and so on.

As a part of the implementation, customers collaborate with the Oracle support and/or development teams to determine the necessary schedule customizations and configuration according to the customer's needs.

POM supports modifying the following attributes of a schedule:

- Enable/Disable a job based on the functional use-case
- Modify the job parameters
- Modify the job dependencies
- Enable/Disable the hourly cycles

5
Notifications

This section provides the list of notifications sent by POM to alert users about events that occur throughout the batch execution.

By default, all notifications are shown on the POM application. It is an option to configure notifications to also send e-mails. This is accomplished through the Oracle Retail Application Admin Console (ORAAC).

1. **Error:** This notification is sent whenever a running batch fails for some reason. This notification is for Oracle internal users.
2. **Warning type:** This notification is sent when application/user is trying to run an already running job.
3. **InformationNotification:** This notification is for System information. This notification is for Oracle internal users.
4. **SystemErrorNotification:** This notification is to notify of all unexpected system exceptions. This notification is for Oracle internal users.
5. **New scheduler day failure:** This notification is sent when there is an issue while setting up the new scheduler day, e.g. previous scheduler day has not yet been completed.
6. **Intraday cycle completed:** This is a low priority notification that is sent whenever an intraday cycle is completed.
7. **Intraday Cycle Skipped:** This is a low priority notification that is sent whenever an intraday cycle is skipped.
8. **NightlyStart:** This notification is sent when the Nightly cycle is kicked off.
9. **Nightly cycle completed:** This notification is sent when the nightly batch completes for the scheduler day.
10. **Long running job:** This notification is sent whenever a job is taking more than the configured threshold time for getting completed.
11. **NightlySummaryReport:** This notification sends out a batch summary report to the configured mail addresses when the nightly cycle is completed. This notification is for Oracle internal users.
12. **IntradayCycleSummaryReport:** This notification sends out a batch summary report to the configured mail addresses when an intraday cycle is completed for each flow.
13. **JosSyncFailed:** This notification is sent out when there is an issue in publishing disabled jobs from POM to JOS.

Note: Disabled jobs are published to JOS as a part of the New Scheduler Day process. If the previous day's nightly cycle is still running at the time of New Scheduler Day creation, then publishing of the disabled jobs to JOS is done when the last job of the nightly cycle completes.

14. **ScheduleChangesSummaryReport:** This notification sends out the Delta Summary Changes Report to the configured mail address. Delta changes include the Jobs added to the current Schedule or Jobs removed or Jobs Status changed from the previous schedule. The External Dependency changes are also shown on the report. If there are no changes between the previous and current schedules, the notification is not sent out.
15. **ExternalDepPending:** This notification is sent when a Job is waiting for and External Dependency.
16. **InterSchedDepPending:** This notification is sent when a Job is waiting for an Inter-Schedule Dependency.
17. **InterSchedDepIssue:** This notification is sent when:
    - The business date between the interdependent Schedules vary by more than a day.
    - The External Schedule is day ahead of the current Schedule and previous day data is not available.
    - The inter-schedule dependencies are not valid and disabled.
    - A schedule includes inter-schedule dependencies or execution links that are not valid.
18. **ApplicationModuleDisabled:** This notification is sent when applications or modules are disabled during MDF synchronization.
19. **ErrorNotificationExternal:** This notification is sent whenever a running batch failed for some reason. This notification is for external customers.
20. **NightlySummaryReportExternal:** This notification sends out a batch summary report to the configured mail addresses when nightly cycle is completed. This notification is for external customers.
21. **ExternalDepComplete:** This notification is sent when an external dependency is completed.
22. **ExecutionLinkIssue:** This notification is sent if there is a failure/warning in the Execution Link invocation.
23. **ApplicationModuleEnabled:** This notification is sent when applications or modules are enabled during MDF synchronization.
24. **ExecutionEngineIssue:** This notification is sent when there are important/critical events/failures with the Execution Engine. For example:
    - When the Job Admin invocation fails or
    - When the request is stuck in SUBMITTING state and not able to auto recover or
    - When the request is stuck in SUBMITTING state with auto reconcile or
    - When the Job is stuck in STARTING state in Job Admin
25. **BatchScheduleImport:** This notification is sent as an update on progress/errors encountered during the schedule import process.

6

User Roles

This section lists the pre-loaded user roles in POM.

Roles	Description
BATCH_ MONITORING_JOB	One of the classic user interface roles. Users within this role are typically retailer administrators responsible for monitoring and executing batch. They can perform select activities on the Batch Monitor screen to move the schedule along.
BATCH_BUSINESS_ JOB	Another one of the classic user interface roles. Users within this role are typically retailer business users responsible for just monitoring batch and configuring POM to enable callbacks into the Company's systems.
BATCH_ ADMINISTRATOR_ JOB	Another one of the classic user interface roles. Users within this role are typically Oracle AMS administrators who monitor, maintain and configure the batch schedules. They also maintain POM application configurations for efficient operations. They troubleshoot batch issues and work with other Oracle development and support personnel to address those issues. Finally they apply POM and batch schedule patches and upgrades. In the new POM user interface, they have additionally been given access to the new AMS Utilities screen.
BATCH_VIEWER_ JOB	Users within this role are retailer business users responsible for just monitoring batch. They have view access to all POM screens except AMS Utilities.
BATCH_SCHEDULE_ CONFIGURATION_ MANAGER_JOB	Users within this role are typically retailer administrators responsible for just monitoring batch and configuring external dependencies and callbacks into the Company's systems. They have view access to all POM screens except AMS Utilities.
BATCH_SCHEDULE_ ADMINISTRATOR_ JOB	Users within this role are typically retailer administrators responsible for maintaining monitoring and executing batch. They have view access to all POM screens except AMS Utilities. They can perform select activities on the Batch Monitor screen to move the schedule along. They also have update access to the Batch Administration screen.
BATCH_ORACLE_ AMS_ ADMINISTRATOR_ JOB	Users within this role are typically Oracle AMS administrators who monitor, maintain and configure the batch schedules. They also maintain POM application configurations for efficient operations. They troubleshoot batch issues and work with other Oracle development and support personnel to address those issues. Finally they apply POM and batch schedule patches and upgrades.

Note: The first three roles mentioned in the above table are associated with POM's classic user interface. They are being deprecated along with the classic user interface. Customers need to migrate to the other four roles before those classic roles are removed.

These roles have been given similar access in the new user interface as the access they had in the classic user interface.

For more information regarding functional access of each POM role, refer to the *Oracle® Retail Process Orchestration and Monitoring Cloud Services Security Guide*.