Oracle Financial Services
Scheduler Services User Guide

Release 23.03.01

F78500-01

February 2023

ORACLE®

Oracle Financial Services Scheduler Services User Guide, Release 23.03.01

F78500-01

Contents

1   Get Help

2   Scheduler Service

3   External Scheduler Interface

1
Get Help

Topics:

1.1 Get Help in the Applications

Use help icons to access help in the application.

Note that not all pages have help icons. You can also access the Oracle Help Center to find guides and videos.

1.1.1 Additional Resources

- Community: Use Oracle Cloud Customer Connect to get information from experts at Oracle, the partner community, and other users.
- Training: Take courses on Oracle Cloud from Oracle University.

1.2 Learn About Accessibility

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

1.3 Get Support

You can get support at My Oracle Support.

For accessibility support, visit Oracle Accessibility Learning and Support.

1.4 Get Training

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

1.5 Join Our Community

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

1.6 Share Your Feedback

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

You can email your feedback to My Oracle Support.

Thanks for helping us improve our user assistance!

1.7 Before You Begin

See the following Document:

- See What's New

2

Scheduler Service

The Scheduler Service is a service that automates behind-the-scenes work that is necessary to sustain various enterprise applications and functionalities. This automation helps the applications to control unattended background jobs program execution.

The Scheduler Service contains a graphical user interface and a single point of control for the definition and monitoring of background executions.

Following are the concepts or terminologies in the Scheduler Service:

- **Batch**: Date and time-based execution of the background tasks based on a defined period during which the resources were available for batch processing.

- **Task:** A batch job is a piece of a program meant to meet specific and business-critical functions. The program is a RESTful API used in a batch.

- **Task Dependency**: When the batch job is submitted, it is moved to the job queue until the system is ready to process. The system process the job based on chronological order or priority in case if more jobs are required to be executed in the job queue.

- **Schedule**: Batch jobs are used to automate the tasks that require to be performed on a regular basis but don't necessarily need to occur during the day or have an employee interacted with the system are batch schedule. Jobs that happen on a regular basis are incorporated into batch schedules.

- **Monitor**: The Scheduler Service enables you to monitor your executions by using a web-browser. It provides real-time feedback on the status of the current encoding job and lists the jobs pending in the batch. You can Cancel or Restart the service when required.

2.1 User Roles

The User Roles and Privileges for Scheduler Service are available in the Users and User Privileges Guide.

2.2 Features in Scheduler Service

The following are the features in the Scheduler Service:

- Use the **Define Batch** feature to create a new batch, modify the batch details, and delete the unwanted batches.

- Use the **Define Task** feature to create tasks for the batches.

- Use the **Schedule Batch** feature to execute a Batch instantaneously and schedule batches.

- Use the **Monitor Batch** feature to track the execution of the batches to view the real time status.

- Use **Scheduler Service Dashboard** to monitor the various activities.

Click the following links for more details:

2.3 Define Batch

Batch is a process of execution of the Date and time-based background tasks based on a defined period during which the resources were available for Batch Processing.

Batch Group is a process of grouping the batches that are required to be execute together for execution of the Date and time-based background tasks based on a defined period during which the resources were available for Batch Processing.

To navigate to the Define Batch Page, complete the following procedure:

1. From the left menu, click **Common Object Maintenance.**

2. Select Operations and select **Define Batch**.

The **Define Batch Page** containing the following details is displayed.

Figure 2-1   Define Batch Page

The Define Batch Page displays the details of all existing Batch like Batch ID, Batch Name, Batch Description, Last Modified By and Last Modified Date. You can also create a Batch Group and execute the Batch Group which has the list of batches that you have selected and grouped.

Table 2-1   Field Description

Field	Description
Batch ID	The Alphanumeric Code assigned to a specific Batch.
Name	The unique Batch Name.
Description	The brief description of the Batch.

Table 2-1   (Cont.) Field Description

Field	Description
Last Modified	The last modified By User, Date and Time details.

2.3.1 Creating a Batch

You can create a new batch in the Define Batch Page, and schedule and monitor the batch that you created.

1. To create a new Batch, click the Add ( ) button in the Define Batch Page.

    The **Create Batch Page** is displayed.

Figure 2-2   Create Batch Page

Enter the **Batch Information** as described in the following table:

Table 2-2   Field Description

Field	Description
Batch Details	

Table 2-2   (Cont.) Field Description

Field	Description
Code	The Unique Alphanumeric Code assigned to the Batch. **NOTE:** • The Batch Name should be unique across the Information Domain. • The Batch Name must be alphanumeric and should not start with a number. • The Batch Name should not exceed 60 characters in length. • The Batch Name should not contain any special characters except "_".
Batch Name	The Batch Name is generated based on the values provided by you. **NOTE:** • The Batch Name should be unique across the Information Domain. • The Batch Name should be Alphanumeric and should not start with a number. • The Batch Name should not exceed 60 characters in length. • The Batch Name should not contain any special characters except "_".
Batch Description	Enter a description for the Batch based on the Batch Name. **NOTE:** The Batch Description should be Alphanumeric. The allowed special characters are_.:- and <blank space>, along with spaces and alpha-numeric. It should not exceed 200 characters in length.
Service URL Name/ Service URL	• Select the Service URL Name from the drop-down list, if it is available. The Service URL is displayed in the **Service URL** field. • To add a new service URL, enter a name to identify it in the **Service URL Name** field and enter the proper URL in the **Service URL** field. You can give partial URL here and the remaining URL in the Task Service URL.

2. Adding Batch Parameters

Figure 2-3   Adding Batch Parameters Window
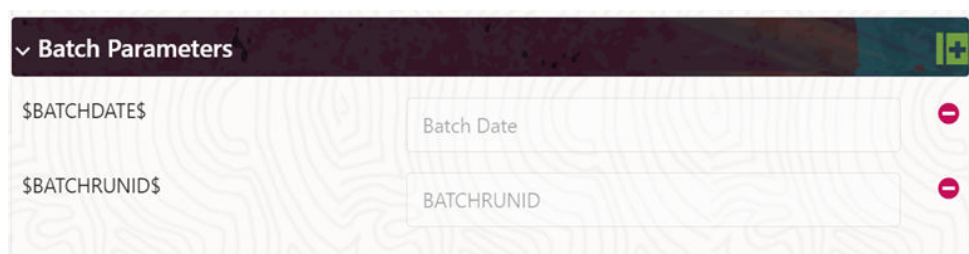
From the Batch Parameters pane, click () to add a new Batch Parameter. By default, $FICMISDATE$ and $BATCHRUNID$ are added as Batch Parameters.

> **Note:**
>
> Enclose the Parameter Value for a Run time with $ symbol. For example, $paramName$

a. Enter the Parameter Name in the **Param Name** field.

b. Enter the Parameter Value in the Param **Value** field.You can delete a parameter by clicking **Delete** corresponding to the parameter.

3. Adding Header Parameters

Figure 2-4   Adding Header Parameters Window
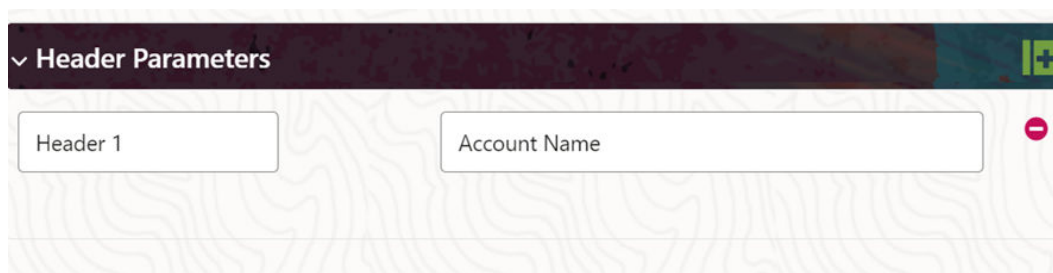
4. Click **Save**. The new Batch is created and displayed in the Define Batch Page.

2.3.2 Editing a Batch

The **Edit Batch** option allows you to edit the Batch details such as Batch Description, Service URL Name and Service URL and also add a new Batch Parameter.

Seeded batches cannot be edited.

1. In the **Define Batch** Page, click (   ) corresponding to the Batch you want to modify.

   The Edit Batch Page is displayed.

2. Modify the required **Batch** details.

   For more information, see Create a Batch section.

3. Click **Save**.

   The edited batch is saved and displayed in the Define Batch Page.

2.3.3 Copying a Batch

The Copy Batch option allows you to copy a Batch that you want to clone or create instances in the system from the Define Batch Window.

To copy a Batch, perform the following steps:

1. In the Define Batch Page, click () corresponding to the Batch that you want to copy. The Copy Batch Page is displayed.

2. Specify the Batch details as you want to clone and copy the existing batch.

   For more information, see Create a Batch section.

3. Click **Save**.

   The copied batch is saved and displayed in the Define Batch Page.

2.3.4 Deleting a Batch

The Delete Batch option allows you to delete a Batch that are no longer required in the system from the Define Batch Page.

Seeded batches cannot be deleted.

To delete a Batch, perform the following steps:

1. From the Define Batch Page, click (✖) corresponding to the Batch you want to delete.

2. Click **OK** in the confirmation dialog box to confirm deletion. If the batch has any active schedules a warning is displayed. On confirmation, all schedules of the batch are also deleted.

2.3.5 Creating a Batch Group

You can create a new Batch Group in the Define Batch Page and schedule and monitor the Batch Group that you created.

To create a new Batch Group, perform the following steps:

1. In the **Define Batch** Page, click ().

   The **Create a New Batch** Page is displayed.

2. Select **Batch Group** option.

3. Specify the following fields:

   - Name

   - Description

   - Add Batches

   > ✎ **Note:**
   >
   > The Add Batches is a multi-select field, you can select the batches that you want to add to the group using this field.

4. Click **Save**.

   The new Batch Group is created and displayed in the Define Batch Window.

2.3.6 Editing a Batch Group

The Edit Batch Group option allows you to edit the Batch Group details such as Batch Group Name, Added Batches, and Batch Group Description.

To modify a Batch Group, perform the following steps:

1. In the **Define Batch** Page, click **Batch Group** option to list the Batch Groups.

2. Click (✏) corresponding to the Batch Group you want to modify. The **Edit Batch** Page is displayed.

3. Modify the required Batch Group details.

   For more information, see Create a Batch Group section.

4. Click **Save**.

   The edited Batch Group is saved and displayed in the **Define Batch Page**.

2.3.7 Copying a Batch Group

The Copy Batch option allows you to copy a Batch that you want to clone or create instances in the system from the Define Batch Page.

To copy a Batch Group, perform the following steps:To copy a Batch Group, perform the following steps:

1. In the **Define Batch** Page, click **Batch Group** option to list the Batch Groups.

2. Click (🗋) corresponding to the Batch Group that you want to copy.

   The **Copy Batch** Page is displayed.

3. Specify the Batch Details as you want to clone and copy the existing batch.

   For more information, see Create a Batch Group Section.

4. Click **Save**.

   The copied Batch Group is saved and displayed in the Define Batch Window.

2.3.8 Deleting a Batch Group

The Delete Batch option allows you to delete a Batch that are no longer required in the system from the Define Batch Window.

To delete a Batch Group, perform the following steps:

1. From the **Define Batch** Page, click **Batch Group** option to list the Batch Groups.

2. Click ( ) corresponding to the Batch Group you want to delete.

3. Click **OK** in the confirmation dialog to confirm deletion.

2.4 Define Tasks

The Define Tasks Page displays the list of tasks associated with a specific Batch Definition. You can create new tasks, edit the existing tasks or delete unwanted tasks. You can also specify task precedence for each task in Task Precedence Window and click the **Add or Remove Precedence** () icon to Schedule the batch.

To navigate to the Define Batch Page, complete the following procedure:

1. From the left menu, click **Common Object Maintenance.**

2. Select Operations and select **Define Task**.
   The **Define Task Page** containing the following details is displayed.

You can select the Batch/Batch Group and select a particular Batch or Batch Group to view the list of tasks associated with that Batch or Batch Group.

2.4.1 Managing Tasks for a Batch or a Batch Group

Batch is a process of execution of the Date and time-based background tasks based on a defined period during which the resources were available for Batch Processing.

Batch Group is a process of grouping the batches that are required to be execute together for execution of the Date and time-based background tasks based on a defined period during which the resources were available for batch processing. You can perform the following operation for the batch based on the task.

You can perform the following operation for the Batch or a Batch Group based on the task.

- Add a Task
- Modify a Task
- Define Task Precedence
- Delete a Task
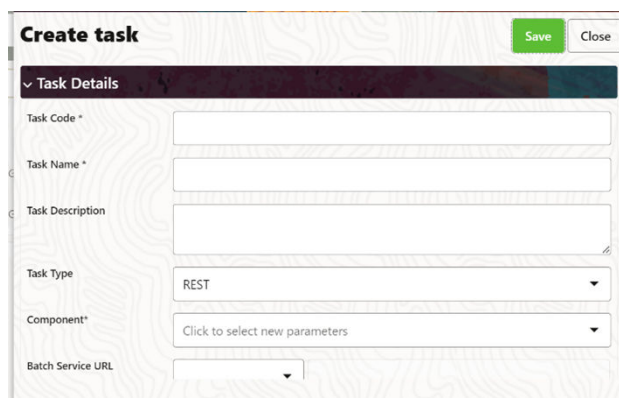
2.4.2 Adding a Task

Adding a new task option allows you to add new tasks to a selected Batch Definition.

To add new task, perform the following steps:

1. Click Define Tasks from the Header panel.

   The Define Task Page is displayed.

2. Select the Batch for which you want to add new task from the Select drop-down list.

3. Click **Add** ( ). The **Create a New Task** Page is displayed.

Figure 2-5   Create Task Page

4. Enter the details as tabulated:

Table 2-3   Field Description

Field	Description
Task Code	
Task Name	Enter the task name. **NOTE:** • The Task Name must be alphanumeric and should not start with a number. • The Task Name should not exceed 60 characters in length. • The Task Name should not contain any special characters except underscore (_).
Task Description	• Enter the Task Description along with spaces and Alphanumeric. No special characters are allowed in Task Description. • Words like Select From or Delete From should not be entered in the Description.
Task Type	Select the task type from the drop-down list.
Batch Service URL	Select the required Batch Service URL from the drop-down list. This can be blank and you can provide the full URL in the **Task Service URL** field.
Task Service URL	Enter task service URL if it is different from Batch Service URL.

5. From the Task Parameters Pane, by default, all Batch Level Parameters are added and enabled as task parameters.

    a. Enter the Parameter name in the **Param Name** field.

    b. Enter the Parameter value in the **Param Value** field.

    You can delete a parameter by clicking corresponding to the parameter.

6. Click **Save**.

2.4.3 Modifying a Task

Modifying a Task option allows you to modify the details of existing tasks of a Batch Definition such as Task Description, Task Type, Batch Service URL and Task Service URL. You can also add a new task parameter and enable or disable already existing task parameters.

To modify a Task, perform the following steps:

1. From the **Define Task** Page, select the Batch whose task details you want to modify, from the Select drop-down list.

2. Click **Edit** ( ) corresponding to the Task whose details you want to modify. The Edit Task Page is displayed.

3. Modify the required Task Details.

    For more information, see Add a Task Section.

4. Click **Save**.

2.4.4 Define Task Precedence

Task Precedence indicates the execution-flow of a Batch. Task Precedence Value facilitates you to determine the order in which the specific Tasks of a Batch are executed.

For example, consider a Batch consisting of four tasks. The first three tasks does not have a precedence defined and hence will be executed simultaneously, during the Batch Execution. However, Task 4 has a precedence value as Task 1 which indicates that, Task 4 is executed only after Task 1 has been successfully executed.

You can set Task precedence between Tasks or define to run a Task after a set of other tasks. However, multiple tasks can be executed simultaneously, and cyclical execution of tasks is not permitted. If the precedence for a Task is not set, the Task is executed immediately on Batch Execution.

To define the task precedence in the Define Task Page, perform the following steps:

1. Click **Add or Remove Precedence** ( ) button corresponding to the task for which you want to add precedence task.
   The Task Precedence Mapping Window is displayed.

   > **Note:**
   >
   > The **Task Precedence** option is disabled if a batch has only one task associated.

   a. Select the batch that you want to execute before the current task, from the Available Tasks pane and click ( ).

   b. To select all the listed batches, click ( ).

   c. To remove a batch, select the task from the Selected Tasks Pane and click ( ).

   d. To remove all the selected batches, click ( ).

2. Click **Save** to update Task Precedence in the batches.

3. Click **Preview** to view the Precedence information.

2.4.5 Deleting a Task

You can remove a task from a Batch Definition which are no longer required in the system by deleting it from the Define Task Window.

To delete a Task, perform the following steps:

1. From the Define Task Page, select the Batch whose task details you want to delete from the Select drop-down list.

2. Click **Delete** ( ) corresponding to the Task you want to delete.

3. Click **OK** in the confirmation dialog to confirm deletion.

2.5 Schedule Batch

The Schedule Batch Page facilitates you to run, schedule, re-start, and re-run the batches in the Scheduler Service. After you upload the data in the required format into the Object Storage, you must load the data into the system using the Scheduler Service. You can schedule them to run in a required pattern and view the Run Time Status of the scheduled services using the Monitor Batch Feature.

Batch is a process of execution of the Date and Time-based background tasks based on a defined period during which the resources were available for batch processing.

You can perform the following operation for the batch:

- Execute a Batch
- Schedule a Batch
- Re-start a Batch
- Re-run a Batch
- Edit Dynamic Parameters
- Task Definition of a Batch

2.5.1 Execute a Batch and Batch Group

The Execute Batch option allows you to run a batch instantaneously. To execute a Batch/Batch Group, perform the following steps:

1. Click **Schedule Batch** from the Header panel.

   The Schedule Page is displayed.

2. Select Batch or Bath Group to execute, from the drop-down list.

3. Select the Batch /Batch Group Name from the Select Name drop down menu. For example, AMLDataLoad.

4. Click **Execute**.

   The **Execution Schedule** Page is displayed.

   Figure 2-6   Execution Schedule Page
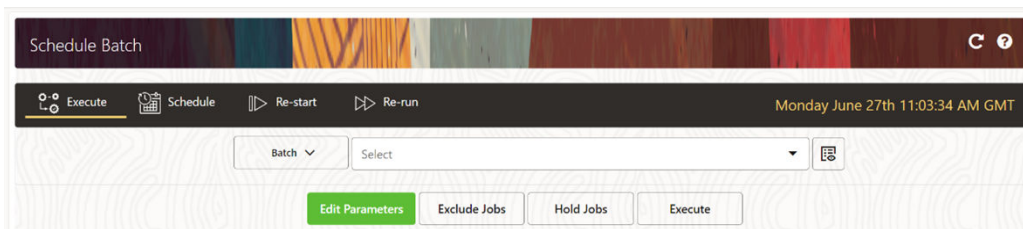
5. If you want to exclude/include some tasks, click **Exclude Tasks**.

6. If you want to hold/release some tasks, click **Hold Tasks**.

7. If you want to edit the dynamic parameters of the batch, click **Edit Dynamic Parameters**.

8. Click **Execute**.

The Batch is executed, and the associated unique execution ID is displayed.

9. Schedule a Batch/Batch Group

You can schedule a Batch/Batch Group to run just for Once, Daily, Weekly, Monthly, or Cron Expression for scheduling the batches. You can also have a user defined schedule to schedule and run a batch.

2.5.2 Schedule Once

To schedule a Batch /Batch Group to run once, perform the following steps:

1. Click **Schedule Batch** from the Header panel.

    The Schedule Batch Page is displayed.

2. In the **Schedule Batch** Page, click **Schedule Once** ().

3. Select Batch or Bath Group to execute, from the drop-down list.

4. Select the Batch /Batch Group Name.

5. Select the Batch or Batch Name you want to schedule for once from the Select drop down menu.

6. Enter a Schedule Name.

7. Select the Start Date on which you want to run the Batch.

8. Click Run Time and select the time at which you want to run the Batch.

9. Click **Schedule**.

2.5.3 Daily Batch Scheduling

To schedule a Batch to run daily, perform the following steps:

1. Click **Schedule Batch** from the Header panel. The Schedule Batch Page is displayed.

2. In the Schedule Batch Page, click **Daily** ( )

3. Select the **Batch /Batch Group Name**.

4. Select the Batch or Batch Name you want to schedule daily from the Select drop down menu.

5. Enter a **Schedule Name**.

6. Select the **Start Date** from which you want to run the Batch.

7. Select the **End Date** till which you want to run the Batch.

8. Select the **Time** at which you want to run the Batch daily.

9. Click **Schedule**.

2.5.4 Weekly Batch Scheduling

To schedule a Batch to run weekly, perform the following steps:

1. Click **Schedule Batch** from the Header panel.

The Schedule Batch Page is displayed.

2. In the Schedule Batch Page, click **Weekly** ( ).

3. Select the Batch /Batch Group Name.

4. Select the Batch or Batch Name you want to schedule daily from the Select drop down menu.

5. Enter a Schedule Name.

6. Select the Start Date from which you want to run the Batch.

7. Select the End Date till which you want to run the Batch.

8. Select the Time at which you want to run the Batch every week.

9. Select the days on a week you want to run the Batch from the Select Days of the Week multi-select drop down menu.

10. Click **Schedule**.

2.5.5 Monthly Batch Scheduling

To schedule a Batch to run weekly, perform the following steps:

1. In the Schedule Batch Page, click **Monthly** ( ).

2. Select the Batch /Batch Group Name.

3. Select the Batch or Batch Name you want to schedule daily from the Select drop down menu

4. Enter a Schedule Name.

5. Select the Start Date from which you want to run the Batch.

6. Select the End Date till which you want to run the Batch.

7. Select the Time at which you want to run the Batch every Month.

8. Select the months in a year you want to run the Batch from the Select Months of the Year multi-select drop down menu.

9. Select the day of the month that you want to run the Batch from the Select Day of the Month drop down menu.

10. Click **Schedule**.

2.5.6 Schedule Cron Expression

To run a Batch in a user-defined schedule, you can have custom schedule with the help of Cron Expression. A Cron Expression is a string comprised of six or seven fields separated by white space. Fields can contain any of the allowed values, along with various combinations of the allowed special characters for that field. For more information, click the icon next to the Cron Expression field.

To schedule a Batch based on Cron Expression, perform the following steps:

1. In the Schedule Batch Page, click **Cron Expression** ( ).

2. Select the Batch /Batch Group Name.

3. Select the Batch or Batch Name you want to schedule from the Select drop down menu.

4. Enter a Schedule Name.

5. Enter the Cron Expression for your schedule.

   For more information about the Cron Expression, click Information icon next to the Cron Expression field.

6. Click **Schedule**.

2.5.7 Re-start a Batch

You can re-start a Batch which has not been executed successfully or which has been explicitly interrupted, or cancelled, or put on hold during the execution process. By re-starting a Batch, you can continue Batch execution directly from the point of interruption or failure and complete executing the remaining tasks.

To re-start a batch, perform the following steps:

1. Click **Schedule Batch** from the Header panel.

   The Schedule Page is displayed.

2. From the **Schedule** Page, select Re-start tab.

3. Select the Batch /Batch Group Name.

4. Select the Batch or Batch Name you want to schedule daily from the Select drop down menu.

5. Select the **Batch Run ID**.

6. Click **Re-start**.

2.5.8 Re-run a Batch

You can re-run a Batch which has previously been executed. Re-run Batch facilitates you to run the Batch irrespective of the previous execution state. A new Batch Run ID is generated during the Re-run process and the Batch is executed as similar to the new Batch Run.

To re-run a batch, perform the following steps:

1. Click **Schedule Batch** from the Header panel.

   The **Schedule Batch** Page is displayed.

2. In the **Schedule Batch** Page, select **Re-run** tab.

3. Select the Batch /Batch Group Name.

4. Select the Batch or Batch Name you want to re-run from the Select Name drop down menu.

5. Select the **Batch Run ID**.

6. Click **Re-run**.

2.5.9 Edit Dynamic Parameters

Dynamic Parameters facilitate you to modify the dynamic parameters for the batch.

You can change the Param Value from the Edit Dynamic Params Page and save the changes to the Batch. The Edit Dynamic Parameters option is available in all the tab in the Schedule Batch Page.

To edit the dynamic parameters for a batch, perform the following steps:

1. In the **Schedule Batch Page**, click **Edit Dynamic Parameters**.
   The Edit Dynamic Params Page is displayed.

2. In the Edit Dynamic Params Page, modify the values as required.

3. Click **Save**. The modified parameters are applied to the Batch.

2.5.10 Task Definitions of a Batch

You can modify the Task Definition state in the Batch Execution Page to exclude or hold the defined task in a Batch from execution. The excluded tasks are therefore assumed to have completed execution and get excluded during the Batch Run.

While executing or scheduling a Batch from the Schedule Batch Page, you can:

- Exclude a task or include the excluded task.

- Hold a task or release the held task.

2.6 Monitor Batch

The Monitor Batch enables you to view the status of executed Batch along with the tasks details. You can track the issues if any, on regular intervals and ensure smoother Batch Execution. A visual representation as well as tabular view of the status of each Tasks in the Batch is available.

To monitor a batch, perform the following steps:

1. Click **Monitor Batch** from the Header panel.
   The **Monitor Batch Page** is displayed.

Figure 2-7   Execution Schedule Page
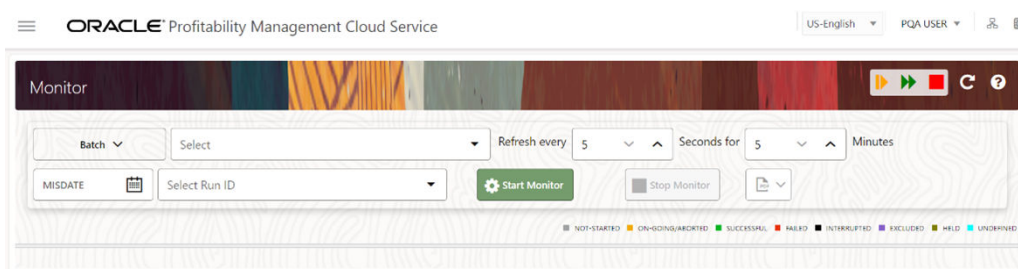
2. Select the Batch/Batch Group from the Select drop-down and then select the Batch Run ID from the Run ID drop-down.

3. Click **Start Monitor**.
   The results are displayed in **Visualization** and **List View** tabs. Details of these tabs are as follows:

   The **Visualization** tab displays the details in the form of a chart represented with the following details:

- **Batch Status**: Displays the batch status, the different batch status are NOT-STARTED, ON-GOING, SUCCESSFUL, FAILED, INTERRUPTED, EXCLUDED, HELD, and UNDEFINED.

- **Batch Start Time**: Displays the Batch Start Time details.

- **Batch End Time**: Displays the Batch End Time details.

- **Task Details**: Mouse-over the task to display its status and details.

The List View tab displays the details in a tabular form with the following details:

- **Batch Status**: Displays the batch status, the different Batch Status are NOT-STARTED, ON-GOING, SUCCESSFUL, FAILED, INTERRUPTED, EXCLUDED, HELD, and UNDEFINED.

- **Batch Start Time**: Displays the Batch Start Time details.

- **Batch End Time**: Displays the Batch End Time details.

- **Task Details**: Mouse-over the task to display its status and details.

- **More Information**: The message returned by the Rest Service.

If you wish to stop the monitoring, select Stop Monitor. You can also specify the Start and Stop Monitor options along with refresh interval in the Refresh every second and minute fields.

> **✎ Note:**
>
> - You can select the refresh interval and the duration for the auto refresh.
>   The refresh interval is defaulted to 5 seconds and duration is defaulted to 5 minutes.
>
>   That is, the refresh happens every 5 seconds for next 5 minutes.
>
> - Range of interval input must be between 5 to 60 seconds and range of duration.
>
> - Input should be between 5 to 180 minutes.
>
> - You can use the Stop Monitor Button to stop the auto refresh.

4. To restart the Batch /Batch Group, select **Restart**.

5. To rerun the Batch/Batch Group, select **Rerun**.

6. To interrupt the Batch /Batch Group, select **Stop**.

7. In the Log Viewer pop-up the log information is displayed. You can click the **Download** icon to download the log or click the **Close** icon to close the log information.

2.7 Scheduler Service Dashboard

To access the Scheduler Service Dashboard, complete the following steps:

1. From the left menu, click **Common Object Maintenance**.

2. Select Operations and select Dashboard.

The Scheduler Dashboard containing the following details is displayed.

Figure 2-8   Scheduler Service Dashboard
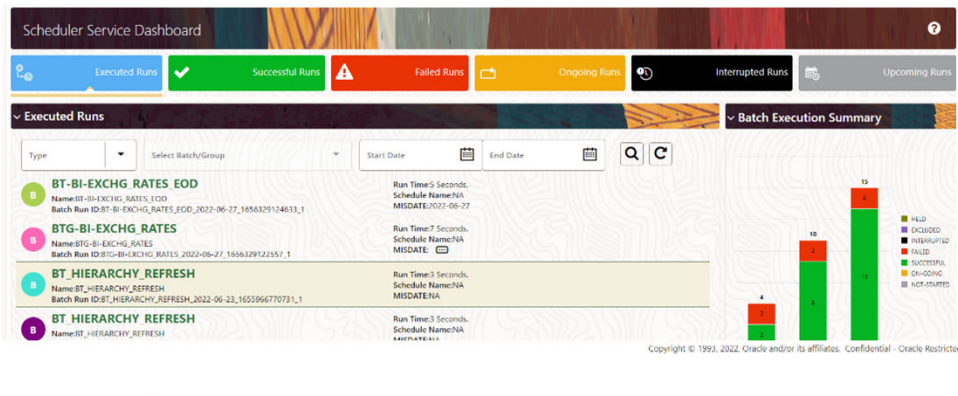
In the Scheduler Service Window, you can view the following details:

The Executed Runs, Successful Runs, Failed Runs, Ongoing Runs, Interrupted Runs, and Upcoming Runs tabs. You can click the tabs to view the details of the Batches based on their status. For example, click Ongoing Runs to view the details of the batches that are currently running.

The Batches that were executed within the last 7 or 30 days contain details such as Batch Name, Batch Run ID, and Run Time. Click 30 days to view the batches that were executed within the last 30 days. You can click the icon corresponding to a Batch to monitor it.

The Batch Execution Summary Pane displays the count of total batches executed that were executed within the last 7 days, 30 days, and 120 days. Additionally, you can see the separate count of successful batches, failed batches, interrupted batches, on-going batches, and the batches which are yet to start, by hovering your mouse the batches.

3

External Scheduler Interface

External scheduler interface help you to execute PBSM tasks using any Command line utility such as cURL commands. You won't require the application interface to execute the PBSM tasks.

You can also integrate the Batches with external schedulers, using the external Scheduler interface.

Related Topics

- Execution API
  The Execution (POST) API triggers a batch or a batch group.

- Execution Status API
  The Execution Status (POST) API provides the current run status of batch/batch group execution.

- Interrupt API
  The Execution Status (POST) API Interrupts a batch/batch group execution.

- Restart API
  The Restart (POST) API restarts a batch/batch group execution.

- Rerun API
  The Rerun (POST) API helps to rerun a batch/batch group execution.

3.1 Rest API Status Codes

Refer to the following table for Rest API Status codes and their description.

Table 3-1   Status Codes

Status Code	Description
0	Success
-1	Failure
-2	Interrupted
1	Not Started
2	Ongoing
3	Aborted
4	Excluded
5	Held
-3	Object does not exist
-4	Invalid arguments passed in request/not enough params in Request body
-5	Invalid request headers/request headers missing
-6	No executable job is present.
-7	Job is already interrupted

Table 3-1 (Cont.) Status Codes

Status Code	Description
-8	Job is not ongoing/aborted

3.2 Execution API

The Execution (POST) API triggers a batch or a batch group.

- **HTTP Method** - POST

- **URL** - `/SchedulerService/rest-api/v1/external/trigger`

- **Header Parameters**

  - **ofs_tenant_id** - Tenant ID of the Application

  - **ofs_service_id** - Service ID of the Application

  - **ofs_workspace_id** - Workspace ID of the Application. It is defaulted to "WS001" and same should be passed each time.

  - **ofs_remote_user** - Used ID of the user. This parameter should be mapped to `'BATCH_EXEC'` function.

  - **locale** - locale in `languageCode-countryCode` format. For example, `en-US`.

  - **Authorization: Bearer <token>** - Access token required to authenticate the API. If this token is not provided, `401 Unauthorized error` is generated. For more information about Bearer token, refer to Generate the Access Token.

- **Sample cURL Command**

```
curl -i -H "ofs_service_id:<Service ID>" -H "ofs_remote_user:<User
ID>" -H
     "ofs_tenant_id:<Tenant ID>" -H "ofs_workspace_id:WS001" -H
"locale:en-US" -H
     "Content-Type: application/json" -H "Authorization: Bearer
<BEARER_TOKEN>" -X POST
     <APPLICATION_BASE_PATH>/<URL> -d '<REQUEST_JSON>'
```

Related Topics

- Batch Execution API
  Use the Execution API to trigger a batch.

- Batch Group Execution API
  Use the Execution API to trigger a batch group.

3.2.1 Batch Execution API

Use the Execution API to trigger a batch.

Attributes

- `batchName` - The unique batch code

- batchType - The object type. For Batch, the batch type should be set to rest.

- dynamicParamList - List of run time parameters which should be overridden over actual values for this trigger. This is an optional parameter.

Request Body

```
{
    "batchName": "<Batch_code>",
    "batchType": "rest",
    "dynamicParamList": {
        "batchParams": {
            "$FICMISDATE$": "<MISDate(yyyy-mm-dd)>"
        }
    }
}
```

Sample Response Body

The following Response body is a sample for Success : 200 OK. For more information about status code in the response body, refer to Rest API Status Codes.

```
{
    "severity": "info",
    "summary": "Object triggered successfully with Run Id:
B2001_2022-05-20_1653041947296_1",
    "batchRunId": "B2001_2022-05-20_1653041947296_1",
    "details": "Object triggered successfully.",
    "statusCode": "0",
    "status": "success"
}
```

3.2.2 Batch Group Execution API

Use the Execution API to trigger a batch group.

- **Attributes**

    – batchName - The unique batch code

    – batchType - The object type. For Batchgroup, the batch type should be set to group.

    – dynamicParamList - List of run time parameters which should be overridden over actual values for this trigger. This is an optional parameter.

Request Body

```
{
    "batchName": "<Batch_group_code>",
    "batchType": "group",
    "dynamicParamList": [
        {
            "batchName": "MIS_BATCH001",
            "batchParams": {
                "$FICMISDATE$": "2022-10-23"
```

```
                }
            },
            {
                "batchName": "MIS_BATCH002",
                "batchParams": {
                    "$FICMISDATE$": "2022-10-28"
                }
            }
        ]
    }
```

Sample Response Body

The following Response body is a sample for `Success : 200 OK`. For more information about status code in the response body, refer to Rest API Status Codes.

```
{
    "severity": "info",
    "summary": "Object triggered successfully with Run Id:
B2001_2022-05-20_1653041947296_1",
    "batchRunId": "B2001_2022-05-20_1653041947296_1",
    "details": "Object triggered successfully.",
    "statusCode": "0",
    "status": "success"
}
```

3.3 Execution Status API

The Execution Status (POST) API provides the current run status of batch/batch group execution.

- **HTTP Method** - POST

- **URL** - `/SchedulerService/rest-api/v1/external/status`

- **Header Parameters**

    – **ofs_tenant_id** - Tenant ID of the Application

    – **ofs_service_id** - Service ID of the Application

    – **ofs_workspace_id** - Workspace ID of the Application. It is defaulted to "WS001" and same should be passed each time.

    – **ofs_remote_user** - Used ID of the user. This parameter should be mapped to `'BATCH_EXEC'` function.

    – **locale** - locale in `languageCode-countryCode` format. For example, `en-US`.

    – **Authorization: Bearer <token>** - Access token required to authenticate the API. If this token is not provided, `401 Unauthorized error` is generated. For more information about Bearer token, refer to Generate the Access Token.

- **Sample cURL Command**

```
curl -i -H "ofs_service_id:<Service ID>" -H "ofs_remote_user:<User
ID>" -H
```

```
        "ofs_tenant_id:<Tenant ID>" -H "ofs_workspace_id:WS001" -H
"locale:en-US" -H
        "Content-Type: application/json" -H "Authorization: Bearer
<BEARER_TOKEN>" -X POST
        <APPLICATION_BASE_PATH>/<URL> -d '<REQUEST_JSON>'
```

Related Topics

- Batch Execution Status API
  Use the Execution Status API to view the current run status of a batch execution.

- Batch Group Execution Status API
  Use the Execution Status API to view the current run status of a batch group execution.

3.3.1 Batch Execution Status API

Use the Execution Status API to view the current run status of a batch execution.

Attributes

- `batchRunId` - Execution Id generated while triggering the object and can be obtained in the response of Execution API.

- `tasks` - List of Task Codes. This is an optional parameter.

Request Body

```
{
    "batchRunId": "<Batchrun_ID>",
    "tasks":["<task_code>","<task_code>"]
}
```

Sample Response Body

The following Response body is a sample for Success : 200 OK. For more information about status code in the response body, refer to Rest API Status Codes.

```
{
    "severity": "info",
    "batchRunId": "BatchTEST1_2022-05-31_1653994545003_1",
    "taskStatusList": [
        {
            "taskCode": "t1",
            "taskStatus": "SUCCESSFUL",
            "statusCode": "0"
        },
        {
            "taskCode": "t5",
            "taskStatus": "FAILED",
            "statusCode": "-1"
        }
    ],
    "batchStatusCode": "-1",
    "batchList": [],
    "batchStatus": "FAILED",
```

```
        "status": "success",
        "statusCode": "0"
}
```

3.3.2 Batch Group Execution Status API

Use the Execution Status API to view the current run status of a batch group execution.

Attributes

- `batchRunId` - Execution Id generated while triggering the object and can be obtained in the response of Execution API.

- `tasks` - List of Task Codes. This is an optional parameter.

Request Body

```
{
    "batchRunId": "<Batchrun_ID>",
    "tasks":["<task_code>","<task_code>"]
}
```

Sample Response Body

The following Response body is a sample for `Success : 200 OK`. For more information about status code in the response body, refer to Rest API Status Codes.

```
{
    "severity": "info",
    "batchRunId": "AbTestBG001_2023-01-27_1674798339245_1",
    "batchStatusCode": "0",
    "batchList": [
        {
            "batchRunId": "AbTestBatch002_2023-01-27_1674798339462_1",
            "batchStatusCode": "0",
            "batchStatus": "SUCCESSFUL"
        },
        {
            "batchRunId": "AbTestBatch003_2023-01-27_1674798339556_1",
            "batchStatusCode": "0",
            "batchStatus": "SUCCESSFUL"
        }
    ],
    "batchStatus": "SUCCESSFUL",
    "status": "success",
    "statusCode": "0"
}
```

3.4 Interrupt API

The Execution Status (POST) API Interrupts a batch/batch group execution.

- **HTTP Method** - POST

- **URL** - `/SchedulerService/rest-api/v1/external/interrupt`

- **Header Parameters**

  - **ofs_tenant_id** - Tenant ID of the Application

  - **ofs_service_id** - Service ID of the Application

  - **ofs_workspace_id** - Workspace ID of the Application. It is defaulted to "WS001" and same should be passed each time.

  - **ofs_remote_user** - Used ID of the user. This parameter should be mapped to `'BATCH_EXEC'` function.

  - **locale** - locale in `languageCode-countryCode` format. For example, `en-US`.

  - **Authorization: Bearer <token>** - Access token required to authenticate the API. If this token is not provided, `401 Unauthorized error` is generated. For more information about Bearer token, refer to Generate the Access Token.

- **Sample cURL Command**

```
curl -i -H "ofs_service_id:<Service ID>" -H "ofs_remote_user:<User ID>" -H
      "ofs_tenant_id:<Tenant ID>" -H "ofs_workspace_id:WS001" -H
"locale:en-US" -H
      "Content-Type: application/json" -H "Authorization: Bearer
<BEARER_TOKEN>" -X POST
      <APPLICATION_BASE_PATH>/<URL> -d '<REQUEST_JSON>'
```

Related Topics

- Batch Interrupt API
  Use the Interrupt API to interrupt a batch execution.

- Batch Group Interrupt API
  Use the Interrupt API to interrupt a batch group execution.

3.4.1 Batch Interrupt API

Use the Interrupt API to interrupt a batch execution.

Attributes

- `batchName` - The unique batch code

- `batchRunID` - Execution Id generated while triggering the object and can be obtained in the response of Execution API.

Request Body

```
{
    "batchName": "<Batch_code>",
    "batchRunId": "<Batchrun_ID>"
}
```

Sample Response Body

The following Response body is a sample for `Success : 200 OK`. For more information about status code in the response body, refer to Rest API Status Codes.

```
{
    "summary": "Execution interrupted successfully for Run Id:
B2001_2022-05-30_1653233511394_1",
    "severity": "info",
    "batchRunId": "B2001_2022-05-30_1653233511394_1",
    "details": "Execution interrupted successfully.",
    "statusCode": "0",
    "status": "success"
}
```

3.4.2 Batch Group Interrupt API

Use the Interrupt API to interrupt a batch group execution.

Attributes

- `batchName` - The unique batch code

- `batchRunID` - Execution Id generated while triggering the object and can be obtained in the response of Execution API.

Request Body

```
{
    "batchName": "<Batchgroup_code>",
    "batchRunId": "<Batchrun_ID>"
}
```

Sample Response Body

The following Response body is a sample for `Success : 200 OK`. For more information about status code in the response body, refer to Rest API Status Codes.

```
{
    "summary": "Execution interrupted successfully for Run Id:
B2001_2022-05-30_1653233511394_1",
    "severity": "info",
    "batchRunId": "B2001_2022-05-30_1653233511394_1",
    "details": "Execution interrupted successfully.",
    "statusCode": "0",
    "status": "success"
}
```

3.5 Restart API

The Restart (POST) API restarts a batch/batch group execution.

- **HTTP Method** - POST

- **URL** - `/SchedulerService/rest-api/v1/external/restart`

- **Header Parameters**

  – **ofs_tenant_id** - Tenant ID of the Application

  – **ofs_service_id** - Service ID of the Application

  – **ofs_workspace_id** - Workspace ID of the Application. It is defaulted to "WS001" and same should be passed each time.

  – **ofs_remote_user** - Used ID of the user. This parameter should be mapped to `'BATCH_EXEC'` function.

  – **locale** - locale in `languageCode-countryCode` format. For example, `en-US`.

  – **Authorization: Bearer <token>** - Access token required to authenticate the API. If this token is not provided, `401 Unauthorized error` is generated. For more information about Bearer token, refer to Generate the Access Token.

- **Sample cURL Command**

```
curl -i -H "ofs_service_id:<Service ID>" -H "ofs_remote_user:<User ID>" -H
     "ofs_tenant_id:<Tenant ID>" -H "ofs_workspace_id:WS001" -H
"locale:en-US" -H
     "Content-Type: application/json" -H "Authorization: Bearer
<BEARER_TOKEN>" -X POST
     <APPLICATION_BASE_PATH>/<URL> -d '<REQUEST_JSON>'
```

Related Topics

- Batch Restart API
  Use the Restart API to restart a batch execution.

- Batch Group Restart API
  Use the Restart API to restart a batch group execution.

3.5.1 Batch Restart API

Use the Restart API to restart a batch execution.

Attributes

- `batchName` - The unique batch code

- `batchRunID` - Execution Id generated while triggering the object and can be obtained in the response of Execution API.

Request Body

```
{
    "batchName": "<Batch_code>",
    "batchRunId": "<Batchrun_ID>"
}
```

Sample Response Body

The following Response body is a sample for `Success : 200 OK`. For more information about status code in the response body, refer to Rest API Status Codes.

```
{
    "severity": "info",
    "summary": "Object triggered successfully for restart with Run Id:
B0001_2022-04-30_1651731208588_1",
    "batchRunId": "B0001_2022-04-30_1651731208588_1",
    "details": "Object triggered successfully.",
    "statusCode": "0",
    "status": "success"
}
```

3.5.2 Batch Group Restart API

Use the Restart API to restart a batch group execution.

Attributes

- `batchName` - The unique batch code

- `batchRunID` - Execution Id generated while triggering the object and can be obtained in the response of Execution API.

Request Body

```
{
    "batchName": "<Batchgroup_code>",
    "batchRunId": "<Batchrun_ID>"
}
```

Sample Response Body

The following Response body is a sample for `Success : 200 OK`. For more information about status code in the response body, refer to Rest API Status Codes.

```
{
    "severity": "info",
    "summary": "Object triggered successfully for restart with Run Id:
B0001_2022-04-30_1651731208588_1",
    "batchRunId": "B0001_2022-04-30_1651731208588_1",
    "details": "Object triggered successfully.",
    "statusCode": "0",
    "status": "success"
}
```

3.6 Rerun API

The Rerun (POST) API helps to rerun a batch/batch group execution.

- **HTTP Method** - POST

- **URL** - `/SchedulerService/rest-api/v1/external/rerun`

- **Header Parameters**

    - **ofs_tenant_id** - Tenant ID of the Application

    - **ofs_service_id** - Service ID of the Application

    - **ofs_workspace_id** - Workspace ID of the Application. It is defaulted to "WS001" and same should be passed each time.

    - **ofs_remote_user** - Used ID of the user. This parameter should be mapped to `'BATCH_EXEC'` function.

    - **locale** - locale in `languageCode-countryCode` format. For example, `en-US`.

    - **Authorization: Bearer <token>** - Access token required to authenticate the API. If this token is not provided, `401 Unauthorized error` is generated. For more information about Bearer token, refer to Generate the Access Token.

- **Sample cURL Command**

```
curl -i -H "ofs_service_id:<Service ID>" -H "ofs_remote_user:<User ID>" -H
     "ofs_tenant_id:<Tenant ID>" -H "ofs_workspace_id:WS001" -H
"locale:en-US" -H
     "Content-Type: application/json" -H "Authorization: Bearer
<BEARER_TOKEN>" -X POST
     <APPLICATION_BASE_PATH>/<URL> -d '<REQUEST_JSON>'
```

Related Topics

- Batch Rerun API
  Use the Rerun API to rerun an existing batch execution.

- Batch Group Rerun API
  Use the Rerun API to rerun an existing batch group execution.

3.6.1 Batch Rerun API

Use the Rerun API to rerun an existing batch execution.

Attributes

- `batchName` - The unique batch code

- `batchRunID` - Execution Id generated while triggering the object and can be obtained in the response of Execution API.

Request Body

```
{
    "batchName": "<Batch_code>",
    "batchRunId": "<Batchrun_ID>"
}
```

Sample Response Body

The following Response body is a sample for `Success : 200 OK`. For more information about status code in the response body, refer to Rest API Status Codes.

```
Success Scenario: 200 OK
{
    "severity": "info",
    "summary": "Object triggered successfully for rerun with Run Id:
B2001_2022-05-30_1653223084727_1",
    "batchRunId": "B2001_2022-05-30_1653223084727_1",
    "details": "Object triggered successfully.",
    "statusCode": "0",
    "status": "success"
}
```

3.6.2 Batch Group Rerun API

Use the Rerun API to rerun an existing batch group execution.

Attributes

- `batchName` - The unique batch code

- `batchRunID` - Execution Id generated while triggering the object and can be obtained in the response of Execution API.

Request Body

```
{
    "batchName": "<Batchgroup_code>",
    "batchRunId": "<Batchrun_ID>"
}
```

Sample Response Body

The following Response body is a sample for `Success : 200 OK`. For more information about status code in the response body, refer to Rest API Status Codes.

```
Success Scenario: 200 OK
{
    "severity": "info",
    "summary": "Object triggered successfully for rerun with Run Id:
B2001_2022-05-30_1653223084727_1",
    "batchRunId": "B2001_2022-05-30_1653223084727_1",
    "details": "Object triggered successfully.",
    "statusCode": "0",
    "status": "success"
}
```