

Plato Orch Archival Instructions

The Archival Framework intends to archive the continuously growing data to the history tables on the basis of frequency and retention period provided by the user. The framework is specific to conductor schema.

The framework supports two levels of Archival. The first level of archival happens from main conductor tables to the corresponding history_1 table and the second level of Archival happens from history_1 tables to history_2 tables. The frequency and retention period for both levels should be maintained, and it should be taken care that the level2 archival has a slower frequency than the level1 archival.

CONFIGURATION

The Configuration required for archiving the data is done in PURGE_CONFIG table -

COLUMN_NAME	DATA_TYPE
1 ID	VARCHAR2(40 BYTE)
2 PURGE_TASK_NAME	VARCHAR2(100 BYTE)
3 FREQ_LEVEL1	NUMBER
4 RETN_PERD_LEVEL1	NUMBER
5 PREV_PURGE_DATE_LEVEL1	DATE
6 NEXT_PURGE_DATE_LEVEL1	DATE
7 FREQ_LEVEL2	NUMBER
8 RETN_PERD_LEVEL2	NUMBER
9 PREV_PURGE_DATE_LEVEL2	DATE
10 NEXT_PURGE_DATE_LEVEL2	DATE

Column Descriptions -

- PURGE_TASK_NAME - This is the name of the configuration which the user want to execute and is sent from the controller in a request for identifying the remaining configuration for purging.
- FREQ_LEVEL1/2 - This is the frequency level where the user can mention the daily, weekly, monthly or annual frequency.
- RETN_PERD_LEVEL1/2 - The retention period signifies the number of day for which the data needs to be retained. For example if the frequency is set to daily and retention period is set for 60 days then in that case the 60 days data will be retained and the purge will be happening daily keeping the 60 days data.
- PREV_PURGE_DATE_LEVEL1/2 - This column gets updated as soon as a purge is triggered by the user.

- NEXT_PURGE_DATE_LEVEL1/2 - The Next purge date level1 also gets updated automatically based on the frequency.

DESIGN

Supported frequencies -

Frequency	Description
1	Daily
2	Weekly
3	Monthly
4	Annually

Important Points to Note:

- Currently the user needs to manually trigger the task for archival. So in that case suppose the next purge date is today and the user did not trigger archiving today and has archived on 3 days later then the purging will be happening on the basis of frequency and the next purge date in the configuration table.
- We have not provided the configuration for tables to be archived. The tables for archival are determined by the framework because most of the tables are dependent on each other and if the user misses a table for archival then there will be data inconsistency.
- After archiving the data, the user might still see the data in base tables from archival dates. This happens when a task of a workflow is still pending, and the workflow is not completed. Only the completed workflows are archived.
- Currently Branch wise archival is not handles. All the data based on the retention period and frequency will be purged (given the workflows are completed).

Plato Batch task details :

Task Name: platoOrchArchivejob

Task Definition: appId:::LMX;microServiceName:::oblm-integration-services;contextRoot:::oblm-integration-services;jobName:::platoOrchArchivejob;purgeConfigName:::<< PURGE_TASK_NAME >>;