TDM V9.3.1 RELEASE NOTES

These Release Notes describe the new features in TDM release V9.3.1 and list bugs that have been fixed since the V9.3.0 release.

- Certification of this TDM release is based on:
  - Fabric 8.2.1 HF2
  - PostgreSQL 17
  - MongoDB extension 1.0.6

MAIN FEATURES AND IMPROVEMENTS

1. Table-level Tasks Enhancements

- Fixing the **filtering** on processed tables when running table-level tasks on **MongoDb**.
- Supporting different source and/or target schema names per environment. The TDM enables now to set Globals in the schema fields of the RefList or TableLevelDefinitions MTables. The Globals can get a different value for each environment (ticket #42245).
- Enhancing the TDM process that brings meta data for the tables to bring the data from the Catalog output for the source environment only. As a result, there is no need to run discovery on the target environment as well (ticket #42408).

2. Integrating the TDM with the Catalog Sequence

- Adding integration of the TDM process to support Catalog-based sequencing. TDM now supports both modes:
  - Implementing the sequence handling in the TDM project. This mode is supported for backward compatibility.
  - Implementing the sequence handling in the Catalog.

    Click here for more information about the Catalog sequence settings.
- Enhancing the TDM templates to add the Catalog Masking actor to load flows. This actor handles both cases if needed – data masking for entity clone (e.g., different name setting for each replica), and sequence replacement (ticket #42006).

3. Task Execution Reports Enhancements

- Ticket #39736 – adding the Statistics Report tab to Extract tasks to enable reporting of extracted record numbers into Fabric for each table.
- Ticket #41184 (which merged into #39736) – adding the number of suppressed (ignored) errors to the Statistics Report tab for a better difference analysis between the number of source and target records on each table.

4. Supporting a Configurable TDM DB Name (tickets #41403, #40088 and #41699)

- Enabling to configure the TDM DB details instead of using hard coded values:
  - TDM now gets the TDM DB details from the TDM interface when creating the TDM DB.
  - Supporting the creation of separate schemas on one central managed PostgreSQL (PG) DB, where each schema serves a separate space, and the schema name is 'k2tdm_<clusterID>.
  - Updating the schema creation command to support both On-Prem and managed PG DBs.

5. Allow Restriction of Testers from Removing Systems and LUs from a Task (ticket #42145)

- Adding a new configuration parameter to TDM to allow restriction of testers from removing systems and LUs from a task. The task LUs will be set based on the task's Business Entity (BE) and environments.
- Click here for more instructions how to restrict testers from removing systems and LUs from a task.

6. Parameters Coupling Mode Enhancement

- Enabling exportation of the LU schema into the TDM DB without creating the FKs in the linked children tables.
- This configurable mode (CREATE_PHYSICAL_FK_IN_MDB_EXPORT_SCHEMA Global) is required when the parent table has a composite PK (i.e., contains a combination of multiple fields) and when only parts of the PK fields are linked to the child LU table. In such case, the PG DB does not enable to create FK on the child LU table.
- Click here for more information about parameters coupling implementation.

RESOLVED ISSUES

- Tickets #42384 and #42487 – fixed the **TDMLUInitBasedOnFabric** flow, which adds the TDM setup to an LU, based on its Fabric structure.
- Ticket #42435 – fixed the error handling of a table-level task in order to prevent loading a table if the table extraction process fails.
- Ticket #42348 – fixed the TDM templates that automatically generate TDM flows. The TDM templates mistakenly identified different sequence names as identical when a sequence name contained another. For example, TEST_SEQ_ID and SEQ_ID.
- Ticket #42284 – fixed the table-level task execution process to support using the DeleteTableByDBCommand TDM flow as a 'customized' flow for specific tables. This

option is needed when all tables of a given interface require a customized delete logic, except for one or a few tables that need to be handed by the default TDM delete flow (DeleteTableByDBCommand).

- TDM portal –
  - Ticket #42112 – fixed the start and end execution datetime on pre and post execution processes.
  - Fixed the Data Type values in the Tasks window.
  - Ticket #42211 – task execution windows - fixed the display of the task execution note.