TDM (TEST DATA MANAGEMENT) UPGRADE PROCEDURE TO V7.5.1

This document describes the following:

How to upgrade TDM onto the present version: from **V7.5 to V7.5.1**

How to re-implement the modified product features.

Notes:

This document does not cover the Fabric server topology changes, such as additions of nodes, data centers, changes of replication factors or consistency level.

The TDM upgrade procedure should be performed on testing environments prior to applying it on your production deployment.

Perform a sanity test upon completion of the upgrade procedure, such as running a few TDM tasks and conducting other checks per the sanity procedure defined in your project.

SOFTWARE UPGRADE PROCEDURE

1. TDM Installation - Prerequisites

The following components should be installed as a prerequisite:

Fabric Server - Fabric 6.5.8 and above for TDM 7.5.1.

PostgreSQL DB - the TDM DB tables are created on a PostgreSQL DB. TDM V7.5.1 supports v9.6 and above (TDM V7.5.1 was certified based on v13).

2. Installation of TDM Package on the Fabric Server

Download Fabric and TDM docker from the download page.

Notes:

It is possible to install the TDM application on several Fabric nodes of a Fabric cluster, but all of them should be working with the same TDM DB.

All Fabric nodes should work with the same Redis instance. This requirement will result in an alignment of the sequence and in masking values generated by the TDM tasks. Open the Redis port on one Fabric node and run Redis.

The TDM and Redis interfaces are defined in the Fabric project.

For more information about TDM V7.5.1 installation see the TDM Installation article in the TDM Configuration.

3. TDMDB Upgrade

Go to $K2_HOME/TDMGUI/createTDMDB/updateTDMDB directory in Fabric.

Set the following environment variables:

export PG_HOST=<postgreSQL DB host>

export PG_PORT=<postgreSQL DB port>

Add execution permissions to **updatek2TDMDB.sh** and run it in order to upgrade the TDM DB.

4. Import the Updated TDM V7.5.1 Library.

Backup the Fabric project. Then do the following steps:

Step 1 - Update the TDM Project Before the Import

- Web Services:
  Delete all the TDM product's files from the Web Services.

- Shared Broadway Flows and Actors:
  - Save a backup of **CustomLogicFlows** actor.

- Shared Translations
  - Edit the trnRefList as follows:
    - Add a new output field after the interface_name field: target_ref_table_name. Set the new field type to text and the length to 50.

Step 2 – Import the TDM Library to the Project

open the TDM Fabric project in Fabric Studio and custom import the following objects of the updated TDM 7.5.1 library into the Fabric project:

- Web Services:
  Import all the Web Services' files from the TDM library.

- Shared Broadway Flows and Actors:
  - Import the shared Broadway flows and actors from the TDM library. Make sure the following actors are not overridden:
    - TDMSeqList
    - TDMSeqSrc2TrgMapping
    - TDMFilterOutTargetTables

- Shared Templates:
  - Import and override the shared templates from the TDM Library.

- Shared Java Files:
  - Import and override the shared Java files from the TDM Library.
  -

- Shared Interface:
  - Import the CASSANDRA_LD (Cassandra loader) interface. This interface is needed to run the **UpgradeRefTables** Broadway flow to recreate the Reference tables in

Cassandra with the updated structure and the Reference data. For more information see section 7 of this document.

- Shared Translations and Globals:
  - Import and override the Shared translations and Globals from the TDM Library. Do not override the trnRefList translation object (already updated by step 1).
    -
- TDM and TDM_LIBRARY LUs:
  - Import and override the TDM and TDM_LIBRARY LUs.
  - Edit the deploy.flow of the TDM LU:
  - Edit the Set Env (FabricSet) actor of the Set Environment stage: Populate the value of the environment key with the environment name from which the deploy.flow needs to get the Cassandra and Redis connection details.
  -

Step 3 – Update the TDM Project After the Import

- Populate the imported **CustomLogicFlows** actor based on the backup file (created before the import). It is recommended to populate the new Description field for each Custom Logic flow to enable the display of the selected Custom Logic flow's description in the TDM GUI task window.

Step 4 – Redeploy the Imported TDM Web Services and TDM:

- Redeploy the Web Services and the TDM LU to Fabric.

5. Update the LUs Schemas

- Regenerate the population of the main source LU table based on the updated populationRootTable.pop.flow template.
- Regenerate the TDMOrchestrator flow on each LU based on the updated template.
- Open the load flows and click the InitiateTDMLoad_Actor actor to refresh it. Save the flows.

6. Deploy the Project Implementation

Redeploy the updated TDM Fabric project to Fabric.

7. Reference Handling

The structure of the Reference Cassandra tables has been changed in TDM 7.5.1.

For more details see https://support.k2view.com/Academy/articles/TDM/tdm_architecture/05_tdm_reference_processes.html.

a. Performance Tuning

The following optional steps can be taken for a better tuning of the Reference's upgrade script execution:

Edit the Fabric config.ini file and add the following section after the default_loader section in order to tune the reference upgrade performance:

```
[Reference_Upgrade_loader]
MODE=TOKEN_AWARE_BATCH
BATCH_SIZE=10000
```

The Reference_Upgrade_loader sets the batch size (# of records) when running the upgrade script to recreate the Reference tables in Cassandra.

For more information about the Cassandra Loader configuration see https://support.k2view.com/Academy/articles/28_cassandra_loader/03_loader_configuration.html.

Open the Environments object and update the Section property of the CASSANDRA_LD interface to Reference_Upgrade. Redeploy the environments to Fabric.

Open the **UpgradeRefTables** Broadway flow. The processing on the tables is done in a parallel mode. By default, the flow processes 5 tables in parallelly. You can edit the threadCount input parameter of the inner flows in order to change the number of parallel tables if needed.

b. Upgrade Script Execution

Run **UpgradeRefTables** Broadway flow (imported from the TDM Library) on your Fabric cluster to recreate the Reference tables in Cassandra with the updated structure and the Reference data. Use the following command:

```
broadway TDM.UpgradeRefTables;
```