TDM (TEST DATA MANAGEMENT) UPGRADE PROCEDURE TO V8.1

This document describes the following:

- o How to upgrade TDM onto the present version: **V8.1.**
- How to re-implement the modified product's 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 8.1 Installation - Prerequisites

Upgrade Fabric to Fabric 7.2 and above.

2. Related Documents

o FABRIC UPGRADE PROCEDURE TO V7.2.0

Note that Step 1 of the Fabric Upgrade Procedure to V7.2.0 document is irrelevant for a TDM project, since the TDM project does not contain the iidFinder process.

For more information about TDM V8.1 installation, please read the TDM Installation article in the <u>TDM Configuration</u>.

3. TDM Upgrade

3.1 Project Code Upgrade

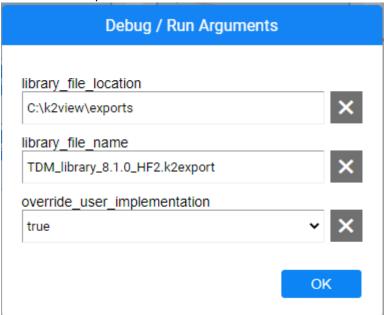
Step 1 – Open the TDM Project in Fabric 7.2 Studio

- Open the TDM project with Fabric Studio 7.2. The Fabric Studio will upgrade the code to the latest version 7.2.
- Right-click the Project Tree > Rebuild All Java Code to verify that there are no compilation errors and that the code was upgraded successfully.

<u>Step 2 – Run the upgradeTDMProject flow</u>

The upgradeTDMProject flow updates the TDM project with the updated TDM library:

- The process backs up the TDM project before it updates it. The backup file is created as a *.tar file in the Projects folder with the following naming convention:
 - <Project name>_bck_<datetime>.tar.gz
 - For example:KB_TDM_FABRIC_PROJECT_bkp_20231108_08h21.tar.gz
- The process replaces the TDM library objects with the updated ones.
- The process updates the LUs in the project with the updated TDM LU tables and populations.
- The process adds the new TDM Globals to the project's Shared Globals.
- Download the TDM library and the upgradeTDMProject export files.
- Open the TDM project in Fabric Studio and import the upgradeTDMProject flow into the Shared Objects (Broadway flow).
- O Populate the input parameters as follows:
 - Library_file_location populated with the full path, of the TDM library export file location.
 - Library_file_name populated with the TDM library export file name.
 - Override_user_implementation set it to true to enable the update of the project's LUs with the updated TDM LU tables.
 - O View this example:



• Run the upgradeTDMProject flow on Fabric debug server. It is recommended to run this flow with a debug OFF mode.

Step 3 – Optional – Update the new TDM Shared Globals

New Globals have been added to the Shared Globals by the upgradeTDMProject flow.
 These new shared Globals can be edited to change their default values if needed:

Global name	Description	Default value	Additional
			information
TDM_DELETE_TABLES_PREFIX	TDM added an automatic generation of the target LU	_TAR	Delete entities implementation
	table to support the deletion		<u>implementation</u>
	of an entity by a TDM task.		
	This Global defines the prefix		
	for the generated target LU		
	tables.		
TDM_BATCH_LIMIT	Limits the number of entities	-1	Task execution
	to be populated into the TDM		process
	execution tables for a task		
	execution. By default, the		
	number of entities for a task		
	execution is unlimited (-1)		
TDM_SEQ_REPORT	When set to true (the default	true	TDM Flows
	value), the task execution		<u>Implementation</u>
	populates		
	the TDM_SEQ_MAPPING table		
	and generates the Replace		
	Sequence Summary		
	Report tab in the task		
	execution report.		
	This Global can be set to false ,		
	thus avoiding these 2 actions		
	for a better performance.		
TDM_POPULATE_JMX_STATS	Set this Global to true to	false	TDM statistics
	enable the execution		handling
	of populationJMX population		
	flow on TASK EXECUTION		
	TDM LU table.		
SEQ_CACHE_INTREFACE	Starting from Fabric V7.2,	DB_CASSANDRA	<u>Fabric</u>
	SQLite and PostgreSQL are		<u>operational</u>
	also supported as Fabric		<u>database</u>
	operational DBs, in addition to		
	Cassandra. If you use the TDM		
	PG DB as the operational		
	(system) DB, update this		
	Global accordingly and set it		
	to TDM.		
SEQ_DO_TRUNCATE	Indicates if the k2masking	false	
	tables need to be truncated.		
SEQ_DROP_KEYSPACE	Indicates if the k2masking	false	
	schema needs to be dropped		
	and recreated.		



Step 4 – Edit the LUs

- Open the LU schema on each LU. Edit the execution order of the LU_PARAMS table population to run after the TDM_LU_TYPE_RELATION_EID table population.
- o If the current version of the TDM project is 7.6 and not 8.0, delete the old LU populations from FABRIC_TDM_ROOT, LU_PARAMS, TDM_LU_TYPE_RELATION_EID and TDM_LU_TYPE_REL_TAR_EID tables. Note that TDM_LU_TYPE_REL_TAR_EID LU table does not need to have an LU population as it is populated by the TDM_LU_TYPE_RELATION table population.

0

Step 5 – Deploy:

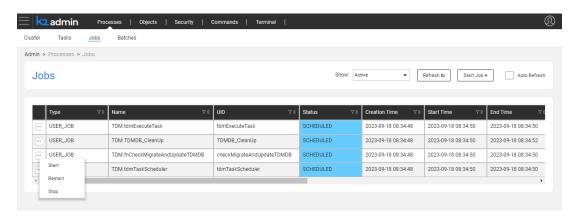
- o Redeploy the Web Services to Fabric.
- o Redeploy the TDM LU to Fabric:
- Verify that the BUILD_TDMDB shared Global is set to false before deploying the TDM LU, otherwise the TDMDB will be created in the deploy flow.
- Redeploy the LUs to Fabric.
- o Redeploy the TDM_Reference to Fabric.

3.2 Run the RunTDMDBUpgradeScripts Flow

- The RunTDMDBUpgradeScripts flow needs to run in order to both upgrade the TDM database and convert the TDM translations into MTables:
 - It runs update_tdmdb_from_v8.0_to_v8.1.sql script. This script adds new fields to the TDM DB tables and populates the new fields in task_execution_entities and <LU Name>_params tables.
 - O It populates the new fields that have been added to task_execution_entity and <LU name>_params tables. This update is needed since TDM 8.1 onwards no longer uses Materialized views that enable an entity selection based on parameters. Instead, new fields have been added to these tables to connect the entity id to its root entity root lu name and root iid.
 - O TDM 8.1 replaces the previous TDM translations with MTables, a replacement that supports the TDM development on both Fabric Studios Desktop-Studio and Web-Studio. The RunTDMDBUpgradeScripts flow runs the Upgrade80_to_81 flow, which in return invokes the convertLuTranslations inner flow in order to convert old TDM translations to the new equivalent TDM MTables. Each execution of the upgrade flow deletes and repopulates the correlated MTables.

TDM UPGRADE FLOW EXECUTION - PRELIMINARY STEPS

- Take the following steps before running the upgrade script:
 - Back up the TDM DB. You can use the pgAdmin and create a backup file on the TDMDB.
 - Deploy all LUs including the TDM LU to Fabric debug server.
 - Stop all TDM jobs on Fabric debug and execution servers in order to avoid locking the TDM DB tables by parallel executions of the upgrade script and the TDM jobs.
 Use the Web Admin to stop the TDM jobs:



TDM UPGRADE FLOW EXECUTION

 Run the RunTDMDBUpgradeScripts flow. Populate the current version and the target version input parameters. Set the target version parameter to 8.1. For example:
 CURRENT_TDM_VERSION = 8.0.

TARGET_TDM_VERSION = 8.1.

UPGRADE THE TDM DB IN THE EXECUTION SERVER

- If you do not have access to the TDM DB from the Studio debug server, do the following to upgrade the TDM DB:
 - Open the Upgrade80_to_81 flow for edit and disable the convertLuTranslations
 Actor.
 - Deploy the TDM LU to Fabric execution server.
 - Open a Fabric console in Fabric execution server and run the RunTDMDBUpgradeScripts flow using the following command:

Broadway TDM.RunTDMDBUpgradeScripts CURRENT_TDM_VERSION='8.0', TARGET_TDM_VERSION='8.1';

Notes:

- The upgrade script updates the <LU name>_params tables based on the task_execution_entities table. By default, the task_execution_entities table contains executions of only the last 7 day (=0.25 month). Therefore, the <LU name>_params table will also contain only the entities of the last 7 days executions (if the related task_execution_entities record is not found, the upgrade job deletes the related <LU name>_params record as well).
- If the <LU name>_param table must contain a history longer than the last 7 days executions, rerun an Extract task on a large population, after the TDM upgrade, in order to repopulate the missing <LU name>_params records.
- The materialized views are no longer needed. Still, to be on the safe side, the upgrade script does not drop the materialized views from the TDM DB. You can drop them after running the TDM 8.1 successfully for a while.

TDM UPGRADE - POST EXECUTION ACTIVITIES

- Deploy the Reference to Fabric (debug and execution servers): the MTables are populated with the translation data.
- Deploy all the LUs to Fabric. Verify that the TDM jobs are up and running.

4. TDM Portal – Re-saving Tasks with Parameters Selection Method

• TDM changes the way the tasks with Parameters selection method are saved in the TDM DB. Therefore, it is required to open and resave TDM tasks with Parameters selection method after upgrading the TDM to V8.1 and before executing the TDM tasks.

5. Data Generation Implementation – PII Fields

- The implementation guidelines for the previous TDM version recommended to use the LU
 population's Masking Actors for the synthetic data generation as well. However, this
 approach was less effective when defining a PII field as an external input parameter for the
 data generation task.
- In order to keep the current implementation where the Masking Actors in the LU population generate the PII values you must check the Mask Sensitive Data checkbox of the Synthetic environment window in the TDM portal.
- Alternatively, you can edit the data generation flows and use either the random data generation Actors or the new TDM 8.1 Actor GenerateConsistent for the PII fields.
- Note: TDM 8.1 supports an integration with Fabric Catalog for data generation and the TDM 8.1 templates create new data generation flows with Fabric Catalog Actors. However, the existing data generation flows can still work. It is not mandatory to edit them.

Click here for guidelines on data generation implementation.

6. Optional - Change Fabric Storage to a Storage that does not Support a TTL

- TDM enables creating tasks with a retention period (TTL) on the task's entities in order to save these entities in Fabric only for a limited period of time. However, if the Fabric storage does not support a TTL for the LUIs (such as PG DB), the TDM needs to limit the TDM task's retention period options to either 'Do not Delete' or 'Do not Retain'.
- Run the following steps to limit the TDM retention period:
 I. Update the tdm_general_parameters TDM DB to limit the TDM task's retention period options to either 'Do not Delete' or 'Do not Retain'.

View the Update statements in https://support.k2view.com/Academy/articles/TDM/tdm configuration/02_tdmdb_general_parameters.html

II. Open the TDM portal, then open the TDM tasks and update them with a retention period other than 'Do not Delete' or 'Do not Retain'.