FABRIC UPGRADE PROCEDURE TO V8.2

This document describes:

- How to upgrade Fabric from the latest **V8.1** to the present **V8.2** version.
- How to re-implement the modified product features.

Notes:

- This document does not cover the Fabric server topology changes, such as nodes addition, DCs, change of replication factor and consistency level.
- It is a must to perform the Fabric upgrade procedure in the testing environments prior to applying it on your production deployment.
- Sanity tests must be performed upon the completion of the upgrade procedure, such as performing a few GET commands and conducting other checks per the sanity procedure defined in your project.

SOFTWARE UPGRADE PROCEDURE

Preliminary Step

Download the latest Fabric package release and copy it to the server(s).

Stop Fabric

Take the following steps in the specified order:

1. If your project has an iidFinder:
   - Stop the iidFinder on all nodes.
   - Wait until Kafka lags are zero in the relevant Kafka topics:
     - Delta_cluster_<LU_name> topics (e.g. deltaGroup)
     - DeltaPriority_cluster_<LU_name> topics (e.g. IDfinderGroupId_Cluster)
   - Run the following command to verify that the lag on the above topics is zero:

     ```
     /opt/apps/k2view/apps/kafka/bin/kafka-consumer-groups --bootstrap-server <internal Kafka server IP>
     --group <Kafka interface group ID> --describe
     ```

   - Investigate the remaining messages in the Delta tables and clean them, by taking the following steps per each LU:
     - Run MIGRATE command on all distinct IIDs.
     - Check the results in order to decide how to proceed with the failed entity messages.
     - Clean the Delta table.
2. Stop Fabric on all nodes.

Open the Package

Perform the following steps:

- Rename the Fabric directory as shown, type the specific Fabric version in the indicated location:

```
cp -r config config_$(k2fabric -version |awk '{print $2}'|head -n1)
mv fabric $(k2fabric -version |awk '{print $2}'|head -n1)
```

- Extract (un-TAR) the Fabric directories from the upgrade package (extract only the directories) as shown. The specific file name will depend on the specific Fabric version:

```
tar -zxvf k2fabric-server-fabric-<package_name>.tar.gz fabric
```

Verify Upgrade Success

Use the following command to verify that the upgrade procedure has been completed successfully:

```
k2fabric -version
```

The result will display the Fabric package number, for example:

```
Tag fabric- 8.2.0_220 at revision = 4c5e23ec2c464daea283faed9bceec2468d21225
```

If there are local files on the server (such as local JARs), their names will be displayed here.

Configuration Changes

Prior to performing the changes, backup your project configuration files, such as config.ini, etc.
Please read the below configuration changes summary per each version.

N/A

Implementation Changes

Please see the below implementation changes summary for each version.

To Version 8.2.1

Modules

- During the upgrade to 8.2.1, remove the **modules** file (located under the config directory) manually. It will trigger a new **modules** file creation during the Fabric upgrade process.
- Note that this manual procedure is only relevant for the upgrade from the versions earlier than 8.2.0.

To Version 8.2.0

Modules

- New modules have been added to Fabric:
  - secretmanager (introduced in 8.1)
  - autoloadsslcontext (introduced in 8.2.0)
- During the upgrade to 8.2.0, it is needed to add the new modules manually.

Fabric Catalog

- It is now mandatory to include the GRAPH_DB_PASSWORD parameter in the config.ini file, as otherwise Fabric would not be able to connect to Neo4j. Until V8.2 this parameter could have remained empty, in which case Fabric would have used the default password.
- The format and language of the Catalog's configuration **discovery.plugins** file has been changed in V8.2. The project-level configuration file is now called **discoveryOverrides.plugins**. It is being created when the settings are updated in the

Catalog's Discovery Pipeline screen. As a result of this change, the projects that have created a project-level **discovery.plugins** file, will need to remove it and re-implement their rules using a new Discovery Pipeline screen.

https://support.k2view.com/Academy/articles/39_fabric_catalog/10_catalog_settings.html

- The DATA_SNAPSHOT_WRITE_BATCH_SIZE_MB parameter has been removed from the config.ini, introducing a new parameter: DATA_SNAP_WRITE_MEMORY_CAP_MB. Thus, if DATA_SNAPSHOT_WRITE_BATCH_SIZE_MB was updated in your project, it should be removed.

  https://support.k2view.com/Academy/articles/39_fabric_catalog/21_advanced_settings.html

Start Fabric

Take the following steps in the specified order:

1. Start the first Fabric node and wait until it has been successfully completed.
2. Start all other nodes.
3. Deploy the project.
4. If your project has an iidFinder, re-start the iidFinder on all nodes.

Note that the above steps may vary per your project's runbook. For example, you may first restart the iidFinder on a single node only, verify that the process has been successfully completed and then proceed to restarting the iidFinder on all other nodes.