



# FABRIC RELEASE NOTES

## FABRIC V7.1.0 RELEASE NOTES

These Release Notes describe the new features in Fabric release V7.1.0 and list bugs that have been fixed since the V7.0.1 release.

Certification of this Fabric release is based on:

- Cassandra version 4.0.3
- SQLite version 3.39.2
- Open JDK version jdk-17.0.3.1
- Kafka version 3.2.0
- Confluent Kafka version 7.2.1
- OrientDB - 3.1.3-tp3 and 3.2.4-tp3
- Elasticsearch - 7.16+ and 8.5.3
- AWS OpenSearch - 1.3.4

## MAIN FEATURES AND IMPROVEMENTS

### 1. MTable

- An MTable is an object created in the Fabric memory upon the deployment, from a CSV file. The purpose of an MTable is to keep reference data as part of the Fabric project and to enable a **fast in-memory** data lookup at run-time. It is recommended to use MTables for small static lists of reference data.
- The data lookup can be performed by one or several MTable keys. The output is the whole MTable row, in additional individual result fields can be returned.
- The search index is created on-the-fly during the first selection, based on the search keys.
- MTables can be used by:
  - New Broadway actors (described further in this document).
  - Various Java methods, to be invoked from Web Services, LU functions, etc.
  - Graphit via a *function* node, by using JavaScript methods (same as Java methods).
- **LIST MTABLE** is a new LIST command, to list the deployed MTables.

[https://support.k2view.com/Academy/articles/09\\_translations/06\\_mtables\\_overview.html](https://support.k2view.com/Academy/articles/09_translations/06_mtables_overview.html)

### 2. Data Manufacturing

- Fabric now separates data generation (manufacturing) from the hashing and caching capabilities (masking). New Broadway actors that support it are described further in this document.

- In addition, Fabric can now generate synthetic data using a flow with the convention `${POPULATION}.population.generator` and the SourceDbQuery actor as described further in this article.

[https://support.k2view.com/Academy/articles/26\\_fabric\\_security/06\\_data\\_masking.html](https://support.k2view.com/Academy/articles/26_fabric_security/06_data_masking.html)

## 3. Broadway

### General

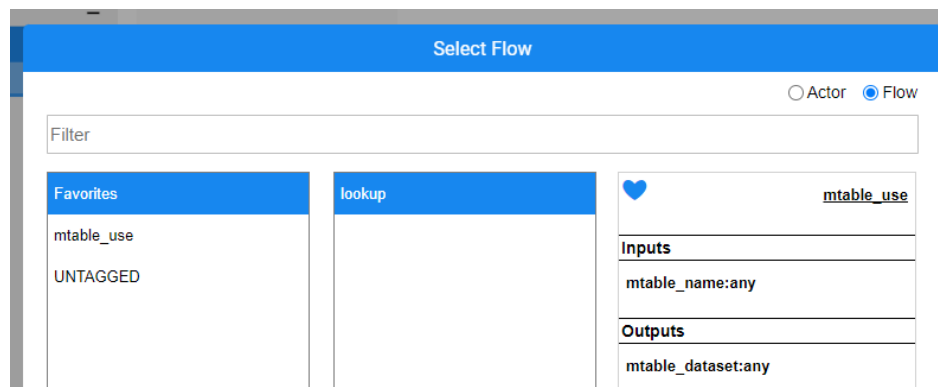
- You can now create a **folder** under Broadway, to better organize your flows. A flow name must be unique across the LU (including the folders).
- You can tag a flow and add a flow description.
  - A flow, created in a folder, is automatically tagged with a folder name. A tag can be updated. Multiple tags per flow are supported.
  - The flow name must be unique across all folders in the same LU.

[https://support.k2view.com/Academy/articles/19\\_Broadway/33\\_flow\\_properties.html](https://support.k2view.com/Academy/articles/19_Broadway/33_flow_properties.html)

- Flow / Actor unification is introduced, providing the following capabilities:
  - An actor can now be invoked from the **InnerFlow** actor. To support that, the drop-down list containing the list of flows is replaced by the popup screen which can be open by clicking on the Expand icon:



- An inner flow can now be added to a flow from the Add Actor screen by switching the radio button from **Actor** to **Flow** in the popup screen and selecting a flow.

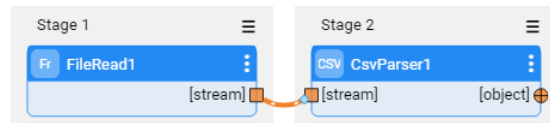


[http://support.k2view.com/Academy/articles/19\\_Broadway/22\\_broadway\\_flow\\_inner\\_flows.html](http://support.k2view.com/Academy/articles/19_Broadway/22_broadway_flow_inner_flows.html)

- The Link Editor has been enhanced with the ability to set the **Link Type**:

Origin Actor	Target Actor	Origin Parameter	Origin Schema Field	Target Parameter	Link Type
DbCommand1	JsonStringify1	result		object	Value

- The link can now be dragged from one port to another, which is indicated by the blue dot:



## Modified Actors

- MaskingSSN** actor continues to receive input to the 'value' field. It recognizes the non-numeric characters and turns them to the delimiter, keeping the same format as the original value. If there are no non-numeric characters in the input, there will be no delimiter in the masked value. For example:
  - The input value = '123456789' will be masked into 987654321
  - The input value = '123-45-6789' will be masked into 987-65-4321
  - The input value = '123&45&6789' will be masked into 987&65&4321
- SourceDbQuery** actor has a new input *rowsGeneratorDistribution*, used to randomize the number of rows per parent row. This input is applicable for Data Manufacturing only.

## New Actors

- The following actors are provided for MTables:
  - MTableLookup**, **MTableRandom** – for fetching data from an MTable by the given key(s) or without any key.
  - MTableLoad** – for either creating a new MTable dataset or replacing an existing one in the Fabric memory.
- [https://support.k2view.com/Academy/articles/19\\_Broadway/actors/09\\_MTable\\_actors.html](https://support.k2view.com/Academy/articles/19_Broadway/actors/09_MTable_actors.html)
- The following actors are provided for Data Manufacturing and Masking:
  - The new **Generators** category (tag) includes a list of actors for generating various types of fake but valid values such as: SSN, email, credit card number. For example, **RandomDistribution** actor generates random values according to the given distribution. The supported distributions are: Normal, Uniform and Weighted. **RandomSSN** actor generates a valid SSN.
  - RowsGenerator** – a new actor for generating random rows given a set of parent rows, a distribution and an inner flow. It relies on the inner flow to generate the actual rows data.
  - Masking** – a new actor to generate random/token data and keep it mapped to input original value. This actor can wrap any data manufacturing actor and add the hashing and caching capabilities on the top of it.
  - Broadway keeps the existing masking actors for backward compatibility.

- **InnerFlowDynamic** actor extends an **InnerFlow** actor. The actor can receive a flow name and a map of input arguments with their values **at run-time**. The output is a map of an inner flow's output arguments and their respective values.
- **SysEnv** actor reads an environment variable, either from the Java properties (-D) or from the system environment. If a variable name is not supplied, the actor returns a map of all Java and system properties.
- **PopulationCount** actor handles the population count and statistics in a population flow that doesn't use the **SourceDbQuery** actor (but rather the DbCommand actor), by populating the **\_k2\_objects\_info** table.

## 4. Reference Tables Backup and Restore

- It is now possible to back up reference tables by schemas or 'ALL' (i.e., defining all schemas for a backup). This capability facilitates the act of either restoring a node if its reference tables data gets out of sync or synchronizing the reference tables data to a newly joined node. The following commands are available:
  - **REF\_BACKUP** is a new command, to back up a snapshot of the specified CommonDB schema and place it into the configured storage.
  - **REF\_BACKUP\_DELETE** is a new command, to delete a backup of the specified CommonDB schema from the configured storage.
  - **REF\_BACKUP\_DOWNLOAD** is a new command, to download a snapshot of the specified CommonDB schema from the configured storage into the specified path.
  - **LIST REF\_BACKUPS** is a new command, to list the ref schema backups.

[https://support.k2view.com/Academy/articles/22\\_reference\(commonDB\)\\_tables/03\\_fabric\\_commonDB\\_runtime.html](https://support.k2view.com/Academy/articles/22_reference(commonDB)_tables/03_fabric_commonDB_runtime.html)

## 5. Graphit

- The Graphit editor offers a quick and efficient way to find a text within the currently opened file, including closed nodes. To do so, you can type the keywords into the search box located at the top toolbar. Once entered, the search box expands into a search result widget, displaying all the occurrences of the searched text.

## 6. Miscellaneous

- The following **new non-DB interface types** are now supported by Fabric and can be used by the Interface Listener to detect new files added to the storage:
  - Amazon S3 Storage
  - Azure Blob Storage
  - Google Cloud Storage



# FABRIC RELEASE NOTES

[https://support.k2view.com/Academy/articles/24\\_non\\_DB\\_interfaces/01\\_nondb\\_interfaces\\_overview.html](https://support.k2view.com/Academy/articles/24_non_DB_interfaces/01_nondb_interfaces_overview.html)

- **SET CLUSTER\_DISTRIBUTE\_AFFINITY = <AFFINITY>** is a new command, to distribute the following command to the specified affinity. Use **ALL** to distribute the following command to all live nodes.
- **LIST BROADWAY\_FLOWS/BF** command can now present not only the flows but also the actors. The command was enhanced with the following flags and capabilities:
  - The population flows are now excluded from LIST BF command's results.
  - [SHOW=FLOW/ACTOR/CORE\_ACTOR/ALL] – when specified, the result includes only the selected type. Default is FLOW. When SHOW=ACTOR, only the project actors are displayed. When SHOW=CORE\_ACTOR, only the core actors are displayed. When SHOW=ALL, both the project's flows and actors (but not the core actors) are displayed.
  - [TAG=NAME] – when specified, the result is filtered by the given tag.
  - [FILTER=<REGEX>] – when specified, the result is filtered by the given expression.
- A new **BATCH\_READ** permission was added. It grants a **read-only** permission on batches (the existing BATCH permission grants read and write permissions). The user with BATCH\_READ permission:
  - Can see and monitor all batches (e.g., via the Batch Monitor screen).
  - Cannot start or retry a batch, cancel batch, change number of workers.
- **Secret Manager Integration** – Fabric now supports integration with Secret Manager services, so that secrets - like passwords, used in interfaces and environments - will not be stored in Fabric itself, but rather their reference IDs. Integration with various external Secret Management providers is supported:
  - AWS Secret Manager
  - HashiCorp Vault
  - Azure Key Vault
- **Web Services improvement** – Fabric now follows RFC recommendation, adopting the "Agent-driven Negotiation" methodology, when a web-service is called: it examines the request accept header and responses accordingly.
  - For example, when using a browser to call to API, Fabric responses with XML because it sends XML as accept header and not JSON.
  - Note that if the accept header is not sent, the default response is JSON, as today.
  - In addition, Fabric now always sends response type header, hinting the request caller about what it gets back.



# FABRIC RELEASE NOTES

## RESOLVED ISSUES

- Ticket 33009 – Customize a message in RejectInstance. The problem has been resolved.
- Ticket 33240 – The Batch monitor shows 100% completion even when there is an instance in sync (status is in process). The problem has been resolved.
- Ticket 33247 – LU Table population is failing on Child tables with sources as Cassandra and having multiple values coming from Parent table. The problem has been resolved.
- Ticket 33394 – The column delimiter does not support recurring characters. The problem has been resolved.
- Ticket 33855 – The Broadway FileWrite actor does not write files bigger than 1KiB to S3 bucket. The problem has been resolved.
- Ticket 33445 – Broadway Population (w/o SourceDbQuery) doesn't update the number\_of\_records entry in \_k2\_objects\_info. The problem has been resolved.
- Ticket 33863 – Partition keys are identified as clustering keys in DbCreateTable actor. The problem has been resolved.