FABRIC RELEASE NOTES

FABRIC V6.5.8 RELEASE NOTES

These Release Notes describe the new features in Fabric release V6.5.8 and list bugs that have been fixed since release V6.5.7.

Certification of this Fabric release is based on:

- Cassandra version 3.11.12 and 4.0.3.
- SQLite version 3.34.0.
- Open JDK version jdk-8u322.
- Confluent Kafka version 5.5.1.
- OrientDB - 3.1.3-tp3 &3.2.4-tp3.
- Elasticsearch - 7.14.1 & 7.8.1.

MAIN FEATURES AND IMPROVEMENTS

1. Fabric Views

- LU views are designed to keep a predefined query on LU tables in the Fabric MicroDB, allowing an easy access to a required data collection. The purpose of creating LU views in Fabric is to enable exposure of either original or manipulated (e.g. masked) data.
    - o LU views are a part of LU definition, but not a part of the LU Schema.
    - o LU views support SELECT statements from one or several joint LU tables.
    - o LU views support the use of LUDB functions.

2. Declarative Field Level Authorization

- Field Level Authorization is a mechanism that enables access restriction to sensitive data, using a Role Based Access approach. Fabric allows exposure of either original or manipulated data to the Fabric user, based on a predefined role, by redirecting the call from querying the LU table to querying its corresponding LU view.
- Field Level Authorization is implemented by creating **security profiles** and by attaching them to the corresponding LU views as follows:
    - o The security profiles are created on Logical Unit level.
    - o For each security profile, the linking between an LU table and a corresponding LU view is defined.
    - o The security profiles can be attached to Fabric user roles, either via the Web Framework Admin or by using Fabric commands. Several security profiles can be attached to one role.

FABRIC RELEASE NOTES

- A new **security_profiles** column has been added to the **roles** table in **k2auth** keyspace, supporting the above mechanism. Please check the upgrade document for the upgrade impact.

- **Elevated Permission** property on Web Services:
  - When a user role has a security profile preventing him from accessing some LU tables, the setting of Elevated Permission to True in a Web Service dismisses this restriction.
  - Using the /*sqlite*/ syntax in a Web Service is possible only when the Elevated Permission property is set to True (the **admin's** permissions). When set to False, the query will be blocked.

3. OAuth 2.0

- **WS Auth&Auth - OAuth Support**: Fabric now supports the standard Open Auth (OAuth) protocol for its Web Services authorization.
  - The expected access token is a JWT. Fabric verifies it using JWKS, which in turn is published by the Auth Server.
  - The access token scopes are mapped to Fabric roles.
  - Fabric provides additional option to set the authorization, that is, by accepting a "client_id" access token claim, which is mapped to a Fabric's APIKEY.
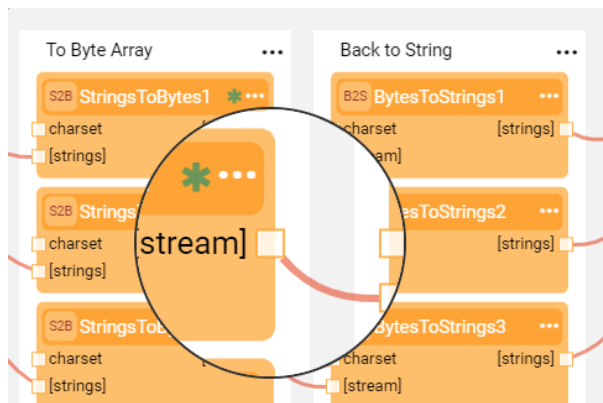
4. Broadway

Usability Improvements

- **Animation** has been added to the connection link between two actors – when the link is selected:
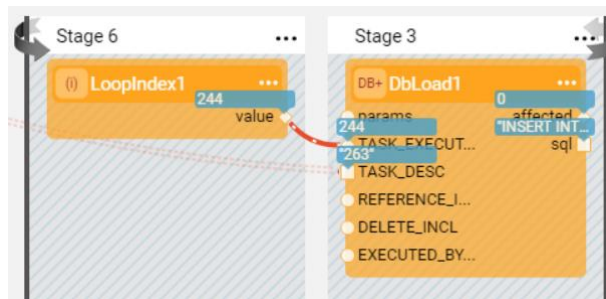
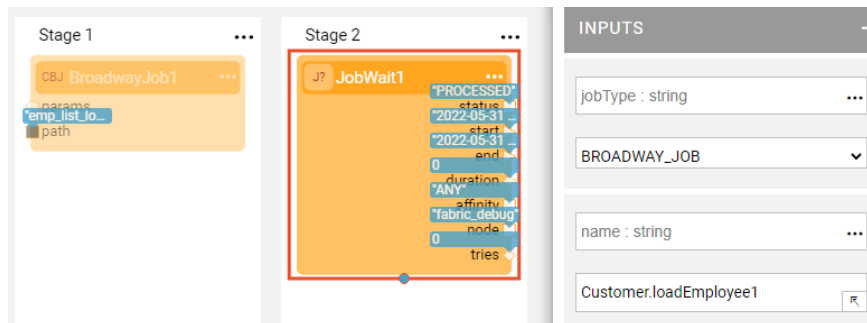- **A Magnifier** has been added, enabling a zoom in on flow:

New actors

- **JobWait** actor allows to wait for Job completion and returns its completion status (PROCESSED). The job completion waiting time can be defined, in a way that if the Job isn't finished on time, a timeout exception is thrown. Scheduled jobs are not supported.

- **JobStop** actor allows to stop a Job and returns its completion status (TERMINATED). The job termination waiting time can be defined, in a way that if the Job isn't finished on time, a timeout exception is thrown.

- **NoTx** actor allows to mark an interface as non-transactional in the context of an active transaction. The actor should be placed in a flow before any action is done on a transaction.

  o The use case for using this actor is the TDM Clone Entity flow. It requires Fabric to be marked as non-transactional interface in order to prevent locking of the Instance SQLite file while the transaction is open on a target DB.

Modified actors

- **InnerFlow, InnerFlowAsync** actors are enhanced as follows: input/output arguments of the selected flow are added automatically to the actor. For **InnerFlowAsync** - only inputs are added.

- **InnerFlow**/**LuFunction** actors refresh was added: If a new argument is added to the flow after the flow was assigned to the InnerFlow actor, clicking on the InnerFlow actor will automatically retrieve and display the new arguments. The same enhancement was added to the LuFunction actor.

- **CsvParser**, **CsvBuilder** actors can support multi-character delimiter. Additionally, **CsvParser** actor can support null values by defining a value to be treated as null as an input. **CsvBuilder** actor can support a record delimiter.

- **MaskingSequence** and other masking actors. When an input argument **onEmpty** is **LEAVE_EMPTY** and the input value is NULL, the actor returns NULL.
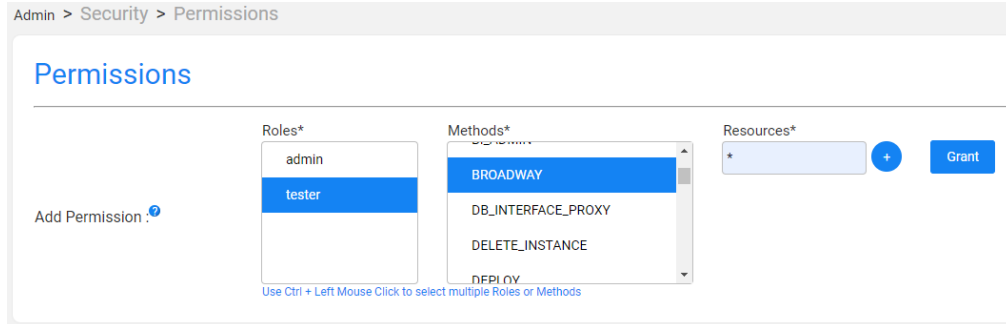
5. Miscellaneous

- **S3** and **Azure Blob Store** have been added to the Fabric Studio as additional LU Storage Types.

- Running the BATCH when the Fabric storage type is AZURE_BLOB_STORE or S3:
  - BATCH process is enhanced. It is now able to run on the LUIs that exist in Fabric when the storage type is defined as AZURE_BLOB_STORE or S3.
  - Now CDC Republish is supported when the storage type is AZURE_BLOB_STORE or S3.

- Kafka CDC Write during the Fabric transaction:
  - Synchronizes the CDC Write process upon completion of the Fabric transaction.
  - This means that the Fabric transaction now waits until the CDC Write process ends.

- HTTP interface - **OAuth 2.0 Client Credentials (Basic auth headers)**:
  - A new authentication type is added.
  - Fabric provides the Client ID and the Client Secret to the authorization server *in the header of the request*.

- The new **[audit_kafka_ssl_properties]** section has been added to config.ini in order to enable defining Kafka SSL properties for the Auditing mechanism.
  - This is applicable only when the Audit mechanism is ON and the persistency strategy is defined as Kafka.

- **Additional Permissions** property on Web Services:
  - Enables activating methods to be accessible by the WS which are restricted for a user who is running this WS (based on the user role definition).
  - The Additional Permission property is available only when the Elevated Permission is set to False.

FABRIC RELEASE NOTES

- o When setting the Additional Permission property, multiple methods can be selected.
- Admin Security Permissions on BROADWAY:
  - o **BROADWAY** method has been added to the list of methods. Adding this method to a role, grants permissions to run BROADWAY flows.

Admin > Security > Permissions

Permissions

Add Permission :❓

Roles*	Methods*	Resources*
admin	DB_ADMIN	*
tester	**BROADWAY**	
	DB_INTERFACE_PROXY	
	DELETE_INSTANCE	
	DEPLOY	

Use Ctrl + Left Mouse Click to select multiple Roles or Methods

- The Fabric commands **VERSION** and **VERSION BASIC** return all relevant basic info about Fabric deployed release:

```
fabric>version;
|Resource         |Result                                  |
+-----------------+----------------------------------------+
|Version          |6.5.8                                   |
|Full Version     |6.5.8_25                                |
|Version's commit |38fc1ce35a0862f1960db8a3d1fdade32a03efaa|
|MDB Version      |3.34.0                                  |

(4 rows)
fabric>
```

- The Fabric command **LIST** was enhanced as follows:
  - o **LIST LUT COUNT=<'Y'/'N'>** - If COUNT = 'Y', additional column is added to the output displaying the num of all instances in Fabric for each deployed LU. If LU_NAME is provided in the command, the output displays one row with the LU name and the number of its instances.
  - o **LIST INSTANCES LU_NAME={NAME}** - List of Instance ID per the given LU name.

RESOLVED ISSUES

- Issue: Prevent a job from taking ownership over two nodes simultaneously and the jobs are running on different nodes, after Fabric restart/redeploy/job restart.
- Issue: When trying to insert a long value into a Decimal column in a Cassandra table, an exception was thrown. Problem is fixed.
- Issue: When trying to insert a **double** value into a **Decimal** column in a Cassandra table, zero was inserted. Problem is fixed.

- Issue: When trying to insert a **double** value into a **BigDecimal** column in a Cassandra table, zero was inserted. Problem is fixed.

- Issue: Implicit conversion from string to Date or Timestamp in an Oracle table was not supported by DbLoad actor. Problem is fixed.

- Ticket 26831 - In Auditing functionality, WS path is not populated in action column of k2_auditing table when a product WS is invoked (such as **/lu/{luName}/{iid}**). Problem is fixed.

- Ticket 28045, 29085 - Tables were not getting populated due to the order of creating the relations in the Schema XML. Problem is fixed.

- Ticket 28806 - Batch process status changed to DONE before all entities processed. Problem is fixed.

- Ticket 28815 - Added a section [delta_kafka_producer] to iifConfig.ini. It used to be added manually. No impact on upgrade.

- Ticket 28980 - CSV Builder actor to support a multi-character delimiter.

- Ticket 28391 - k2profiler creates the snapshot right after getting the user command, without any delay. Additionally, the error caused by concurrency of creating the snapshot and closing the profiler is fixed.

- Ticket 28797 - REF_SYNC_WAIT showed that the transaction is completed without waiting for the commit completion in SQLite. Problem is fixed.

- Ticket 28896 - IIDF not able to process Kafka messages if table/field names were in lowercase. Problem is fixed.

- Ticket 28959 - Selects are failing on the same session of the GET with empty schema in enrichment. Problem is fixed.

- Ticket 28657 - When editing a DB Query (regular population) or a DbCommand actor (flow) that has a /*sqlite*/ hint, the hint is removed from the query by the Query Builder. Problem is fixed.

- Ticket 29010 - Admin UI Data Explorer fails to update a record. Problem is fixed.

- Ticket 29013 - Support importing/dragging items from REF to LU during Custom Import.

- Ticket 29059, 29060 – Fabric implementation compilation fails but the build succeeds. Problem is fixed.

- Ticket 29071 - Number of failed entities is displayed incorrectly in the Batch Monitor, in multi DCs. Problem is fixed.

- Ticket 29135 - CDCDataPublish starts with configuration of Consumer. Problem is fixed.

- Ticket 29152 - Missing Refresh of the **function name** input argument of the LuFunction Actor when calling an Actor in loop. Problem is fixed.

- Ticket 29083 - For Cassandra interface: the **consistency** and **serial consistency level** parameters could be set using the server connection string.
  - For example: consistency=QUORUM&serialConsistencyLevel=SERIAL.
- Ticket 28196 - Connections pooling Fabric are not released and remain idle. The problem was in the Stats mechanism updating the pool state without any actual connections remaining open. Problem is fixed.
- Ticket 29298 - <IIDF> config changes.
  - Add a visible flag in iifConfig.ini FAIL_PARSING_ID - default set to false. The code logic is unchanged on false. On true, if parser id fails to send a replicated message, an exception is thrown and as a result iidFinder stops.
  - Change IID_TOKEN_BINDER default to '_@_' Please check the upgrade document for the upgrade impact.
- Ticket 29102 - Error on building a Data Catalog when the SourceDbQuery actor in a population doesn't have the same output column names as the DbLoad actor's input. Problem is fixed.
- Ticket 28480 - ExagoWebApi URL was hardcoded in Fabric to http, regardless of the Fabric running mode (http or https). Problem is fixed.
- Ticket 29007 – Admin UI inconsistency in Batch Monitor screen showing the Batch Completion = 100% but the migrate status is IN_PROGRESS. Problem is fixed.
- Ticket 28492 - Studio fails to deploy because of .DS_Store files. Problem is fixed.
- Ticket 28797 – Common table updates from LuFunctions inside Broadway flow not working as expected. Problem is fixed.
- Ticket 27892 – Java API to get LUT table fields does not preserve the order. Problem is fixed.
- Ticket 28021 – Low API Response Rate Fabric. Problem is fixed.
- Ticket 29238 – Ability to set a min size of transaction to be considered as long transaction. By default, it equals to TRANSACTION_BULK_SIZE.
- Ticket 29215 – Fabric node not starting or not stopping. Problem is fixed.
- Ticket 28959 – Selects are failing on the same session of the GET with empty schema in enrichment. Problem is fixed.
- Ticket 29224 – Even when running migrations with set local true, the partition job must be up else instances are failing. Problem is fixed.
- Ticket 29347 – iidFinder partition - set workers count on LU level is required. Problem is fixed.

FABRIC RELEASE NOTES

- Ticket 29463 – Studio fails to tryLocalFastDeploy during save iidFinder XML. Problem is fixed.

- Ticket 29337 – When setting the QUEUES_CLEANER_DELAY_MS parameter to anything less than 1000, the error is thrown. Problem is fixed.