



FABRIC RELEASE NOTES

FABRIC V6.5.7 RELEASE NOTES

These Release Notes describe the new features in Fabric release V6.5.7 and lists bugs that have been fixed since release V6.5.6.

Certification of this Fabric release is based on:

- Cassandra version 3.11.9 and 4.0.1.
- SQLite version 3.34.0.
- Open JDK version jdk-8u312.
- Confluent Kafka version 5.5.1.
- OrientDB tp3-3.1.2.
- Elastic oss-7.6.0.

MAIN FEATURES AND IMPROVEMENTS

1. iidFinder

- Added a configuration to ignore null primary key messages by just reporting to log and skipping to the next message. Two new parameters were added to the iifConfig.ini file:
 - SKIP_CORRUPTED_MESSAGES – set to FALSE by default. If set to TRUE the new functionality will start to work.
 - MAX_SKIPPED_CORRUPTED_MESSAGES – maximum number of skipped corrupted message. Default is to 10,000. To set an unlimited number, enter -1 .
- Any change on these two parameters requires an iidFinder restart.

2. Azure Blob Store

- Performance improvements when using Azure Blob Store as a Fabric storage layer (less API calls on read).
- Better optimistic locking support.

3. Transition Storage – Experimental Feature

- A new storage type was created that can be used to move data *from* storage A *to* storage B (e.g. Cassandra to S3) gradually with no system downtime or heavy resync processes.
- The following configuration was added in config.ini file under the fabricdb section:



FABRIC RELEASE NOTES

- `MDB_DEFAULT_SCHEMA_CACHE_STORAGE_TYPE=TRANSITION`
- `STORAGE_TRANSITION_FROM=CASSANDRA`
- `STORAGE_TRANSITION_TO=S3`
- When trying to bring data from a Fabric microDB, Fabric first checks (ie, reads) `STORAGE_TRANSITION_TO` for data. If Fabric cannot find data there, it then reads `STORAGE_TRANSITION_FROM` for data.
- On save, Fabric saves data in `STORAGE_TRANSITION_TO`. If the microDB was found in `STORAGE_TRANSITION_FROM` during the read process, then after saving the data in `STORAGE_TRANSITION_TO`, Fabric deletes the data that was found in `STORAGE_TRANSITION_FROM`.
- If `STORAGE_TRANSITION_FROM` is set to blank, the system will stop reading `STORAGE_TRANSITION_FROM` (no need for restart).

RESOLVED ISSUES

- Ticket #28192 – Substring Actor cannot get last letter.
- Ticket #28154 - In overloaded environment, where there are many cdc jobs (publishers) that might be joined and/or node is overloaded, a Kafka rebalance will eventually occur and it reports errors.
- Ticket #27994 – Any Fabric node restart or deployment triggers an LU version change when the implementation includes CDC fields.
- Ticket #28161 – Version basic command returns incorrect value and improving version/version info commands.
- Ticket #28105 – GET with parallel is failing. Improve the way Fabric handles sync off with a transaction. In this case, Fabric will not create the empty db and fail the iid.
- When running a GET and ROLLBACK at the beginning of a transaction (which should trigger a schema change), the ROLLBACK will not roll back the schema, and if the microdatabase is in cache, the microdatabase will not upgrade.
- Ticket #27981 – SEARCH command should return the fields order according to the order defined in the implementation.
- Drop k2_ref failed when long transaction or a snapshot was still running.
- Fail to reconnect – SEARCH session. When a session is opened to elasticsearch, it will never recover if elasticsearch is down temporarily.



FABRIC RELEASE NOTES

- Search loader misses drop schema message. If elasticsearch is down while a drop luetype occurs, then when elasticsearch goes up again, the search loader misses the drop schema message (will not be published to elasticsearch).
- Fabric BI application was fixed to support Fabric in hardening mode.
- Ticket 28045 - GET fails when there is a population connected with two or more columns and one of the columns is an INTEGER and the integer column is not the first column.
- Ticket #26567 – improve Broadway Assert* actors to involve Actor input parameters in the Actor text message.
- Ticket #28370 - Enabling auditing functionality while using Kafka as the persistence strategy is not working.
- When using CDC functionality in OFF mode sync process is getting stuck when Kafka is not available.
- Ticket #28299 – improve CDC memory consumption.
- Ticket #28419 – common tables, fix retry mechanism to work without dependency on the bulk size.
- Ticket #28452 – Null Pointer Exception in CDC_TRANSACTION_PUBLISHER job.
- Ticket #28390 - transactions reported as dropped by CDC at the time of CDC jobs start.