FABRIC RELEASE NOTES

FABRIC V8.2.3 RELEASE NOTES

These Release Notes describe the new features in Fabric release V8.2.3 and list bugs that have been fixed since the V8.2.2 release.

Certification of this Fabric release is based on:

- Cassandra version 4.1.8
- SQLite version 3.46.1
- OpenJDK Runtime Environment 21.0.3
- Confluent Kafka version 7.6
- Neo4j 5.25.1 – enterprise
- Elasticsearch – 8.5.3
- AWS OpenSearch – 2.17
- PostgreSQL 15.12

MAIN FEATURES AND IMPROVEMENTS

- Graphit
  - Added the ability for nested Graphit calls. Graphit can now call another Graphit. A new node type named *include* is introduced; the 'code' should contain the Graphit name with/within the folder name.
  - In cases where an XML format contains values and attributes, we can now support a field calculated by a query/Broadway that returns a single result (setting needs to be one=true) while still having child attributes.
- PS command throws a null pointer exception when running following batch process cancellation.
- Ticket #42361 – when running a batch retry due to some failed entities, the overall number of handled entities is reduced by one (i.e., entity count mismatch).
- Ticket #42791 – jobs mechanism is not using logical id functionality correctly when defined within the Fabric cluster.
- Ignoring the URL token when calling a Fabric Web-Service. This can be configured in the config.ini file by setting the DISABLE_LEGACY_TOKEN parameter to 'TRUE' under the fabric section. By default, this parameter is set to 'FALSE'.
- Adding the word 'following' to the list of the reserved words in SQLite.
- INTERFACE_LISTENER job is not triggered when adding/deleting a file in S3, GCP or Azure storage.
- Ticket #42849 – iidFinder – failing to process records on spaces that are configured with AWS KEYSPACES.

- Ticket #42935 – adding required jars to the Fabric package in order to connect to Azure SQLServer with ActiveDirectoryManagedIdentity.

- Experimental feature - Introducing two new actors named DbFlushBatch and DBErrorHandlerBatch, which flushes the current batch, returns actual success and affects rows. It ensures errors can be handled without failing the entire batch.

- 3rd party security vulnerabilities resolved.