FABRIC RELEASE NOTES

FABRIC V6.5.0 RELEASE NOTES

These Release Notes describe the new features in Fabric release V6.5 and lists bugs that have been fixed since the latest V6.4.

Certification of this Fabric release is based on:

- Cassandra version 3.11.9.
- SQLite version 3.27.2.
- Open JDK version jdk-8u252.
- Confluent Kafka version 5.5.1.
- OrientDB tp3-3.1.2.
- Elastic oss-7.6.0.

MAIN FEATURES AND IMPROVEMENTS

1. FIPS Certification

- Fabric encryption algorithm can now be set as FIPs certified, implemented by Bouncy Castle FIPS Java API 1.0.2.1, holding CMVP Certificate #3514.
- It is possible to turn the FIPS certification module ON or OFF, within Fabric modules file, by adding fips:mode=on (by default it is set to off).
- A new command was introduced called **version fips** that returns the FIPS version and status.

2. User Identification and Access Management (IAM)

Fabric can now provide user identification and access management (IAM) for Web, Console and Web Services access, either using Fabric local repository or by using the organization's identify provider (IDP) which Fabric is integrated with. This feature, although enabled in V6.5 must be explicitly configured for it to be operative.

Fabric supports several Identity and authentication providers:

- **Fabric**, using its repository (default).
- **LDAP**, via LDAP integration.
- **ADLDAP**, via Active Directory (AD) LDAP integration.
- **SSO vis SAML**, for Web and WS access. Done via SAML IDP integration.

  Fabric is ready to work with commonly used and major IDPs such as Azure and Okta.

Fabric also provides the option to block the access, empowering the security access control, for example when users' access is limited only to specific node(s) in cluster.

Features and capabilities:

- Fabric enables to use a proprietary custom authenticator, when required.
- This feature is flexible, enabling to define a **sequence of authenticators**, each is used as fallback to its predecessor.
- This feature provides the flexibility to have a per channels authenticator. For example, web-apps might use access via SAML, where console access uses LDAP.
- Data protection:
    - All supported transit methods provide secured SSL/HTTPS access.
    - Fabric, and IDP SAML's intersections are done using certification and encryption methods.
    - Users, their credentials, and their association to roles are not stored in Fabric, this allows the organization full control over access capabilities.
    - LDAP admin user credentials are encrypted at the Fabric configuration.
    - Users' login access to Fabric is recorded into the Fabric **Auditing** mechanism with the information about the channel and which authenticator was used.
    - **Web services Security**: now Fabric supports *bearer header authorization*, where the JWT (JSON Web Token) originated and is sent securely by the API client as the *bearer* token. In this case the user's authentication process is not done at Fabric, allowing an application that already authenticated a user to delegate API calls by providing user's information. Fabric validates that this token can be trusted and uses it to give access permissions to the API's client.

3. Broadway

Modified Actors

The following Actors were modified:

- **Email** – now supports the email body in HTML format and can send an attachment (one or many).
- **BatchWait** – an interface input argument was added. By default, it is set to **fabric** and can be changed to remote Fabric Interface.
- **DbCommand, DbLoad** and other DB Actors – now support the **batch** mode. When set to true, the Actor accumulates statements and performs them as a batch for better performance. It needs to be run in a transaction and the errors are reported as the batch is committed (every 1000 record or on commit).

New Actors

The following new Actors were added:

- **Unzip** – unzips data from a zip file.

- **ToString**, **ToBuffer**, **ToNumber** and **ToBool** – turns the input into a String, a byte[] buffer, a Number or a Boolean using the standard Broadway conversion rules.
- **LoopStop**, **LoopSkip** – skips the current loop iteration and goes to the next one, or stops the loop.
- **ImageLoad**, **ImageSave**, **ImageText**, **ImageScale**, **ImageDuplicate** and **ImageCreateCheck** - image manipulation actors.

Usability Improvement

The Broadway UI usability was improved by:

- Changing the color of the **Stage Condition** actors and the **ELSE** mark to green.
- Aligning the layout of all actor types in disabled mode – to make the badge color grey and the actor's name opaque.

Profiler

Broadway can now add the profiler results to the flow output, including the breakdown per Stage / Actor / Iteration. The profiler results are available either via the Run Results window in the Fabric Studio or as part of the **broadway** command results.

Trace and Statistics

Broadway is now included into the Fabric Statistics (JMX metrics) and the Trace mechanism with performance metrics per Flow / Stage / Actor / Iteration.

4. iidFinder

Consumer Group ID

The ability to configure the Consumer Group ID according to the required naming convention was added:

- A new parameter IIDF_CONSUMER_GROUP_ID was added to iifConfig.ini.
- The parameter holds the default Consumer Group ID prefix **IDfinderGroupId** with the cluster ID, as follows: IDfinderGroupId_${cluster_id}.
- In the customized setting, the prefix can be updated to the required value.

Removed: cacheTableLuAlias Property

- <cacheTableLuAlias> property is removed from IID Finder XML.

Removed: Proactive Indicator from DB interface

- Proactive Indicator (which defines whether Proactive Sync should be enabled) was removed from the DB Interface definition and will now be supported on LU Table Population only.

Removed: staging.xml

- Staging.xml was removed and is not supported anymore. Currently only IID Finder XML generated from Fabric Studio is supported.

Important: Refer to Fabric_Upgrade_Procedure_To_V6.5.pdf document for the description of required implementation changes.

5. Fabric Studio

Studio Breakpoints

A new panel was added to the "Server / Activity Logs" area, which displays a list of the breakpoints in the open Fabric objects (currently only Broadway flows are supported), including the File name (flow name) and the ID (e.g. stage where the breakpoint is set).

The panel is refreshed every 1 sec, but only when the panel is open (visible to the user). The yellow arrow sign on the panel indicates the breakpoint where the flow currently stopped. Double-click on the breakpoint description in the panel brings the related flow to the front.

Known limitation: if there are flows in different LUs with the same name, the double-click will open the first flow that it finds with that name (not necessarily the correct one).

Local Deployment Improvement

Fabric performs local deployment according to the following logic:

- If the Schema was changed, full deploy is performed.
- If Java was changed, only Java resources and Broadway are deployed.
- If Broadway was changed, only Broadway is deployed.

6. Miscellaneous

Audit

The Auditing configuration in config.ini has been refactored by adding the new sections **[audit]** and **[audit_kafka_producer]**. All the relevant properties were moved to these new sections from either **[fabric]** section or from the common area.

The following enhancements and fixes were done to the Auditing mechanism:

- The user login & authentication to the Web Framework is now audited, as follows:
  - Action = LOGIN, Protocol = HTTP/1.1, Query = LDAP/SAML/FABRIC
- There are no additional LOGIN audit entries when the user views or makes actions on the Web Framework.
- User login to the Fabric console is now audited, as follows:
  - Action = LOGIN, Protocol = DRIVER, Query = LDAP/FABRIC

- Failure logins are audited as well. Note that the actual login method is populated at Query. Thus if LDAP is the first authenticator and it failed and then the next one is local (Fabric) and it passed, only the last entry Query = FABRIC is captured.
- Logout is not audited.

Extend Fabric JDBC API: Support for SET Command via Connection URL

Fabric now supports the ability to set the session variables via the Fabric Connection URL using the following syntax:

- jdbc:fabric://[server:port]?user=[user_name]&password=[password]&[key1]=[value1]&[key2]=[value2]
- For example: jdbc:fabric://localhost:5124?user=admin&password=admin&timeout=0&sync=off

This feature can be used when external systems (such as BI systems) that are not familiar with the Fabric syntax, are connecting to Fabric.

Query LUI Without GET ("No GET")

Fabric now has the ability to query a Logical Unit without performing an explicit GET command when the following conditions hold:

- Set AUTO_MDB_SCOPE = true.
- The SQL statement includes a WHERE clause with the filter by IID.

This feature enables querying Fabric by various external systems (such as BI) that are not familiar with the Fabric syntax. They can use standard SQL language rather than the Fabric GET command. For external connection to Fabric, AUTO_MDB_SCOPE=true should be concatenated to the Fabric connection string.

Sync On Protection: Change in Failure Handling

The failure handling behavior of the Sync On Protection mode was changed as follows:

- If there are several sync requests for the same LUI on the same node within the Protection period and the first GET fails, all the remaining requests will fail with the same reason without trying to sync again.

Before 6.5, the behavior in the above scenario was to try and sync other requests again.

VERSION command

The VERSION command was enhanced as follows:

- **version info** or **version details** or **version**, returns the Fabric package name, MDB version and a list of changed or updated files.
- **version basic** command, returns the Fabric version only, for example 6.4.2 or 6.5.0.

LIST USERS command

The LIST USERS command enables filtering the retrieved list either by specifying the required role or user.

- Filtering by role is done by adding the argument `role_filter=<filter>`. The result will show only users whose role match the role defined in the filter.
- Filtering by user is done by adding the argument `user_filter=<filter>`. The result will show only users whose user name match the user name defined in the filter.

The command filtering supports exact match as well as similarity filtering by using wildcards.

Fabric Studio and Web Framework Rebranding

Rebranding was done on the Fabric Studio and Web Framework aligning their look & feel with K2View's logo, colors, fonts, etc.

Parallel Load Big LUI from Cassandra - Experimental

Performance improvement when loading Big LUIs has been accomplished by carrying out a parallel load from Cassandra. The memory to be used for the parallel load is set by configuration, as follows:

- #ASYNC_LOAD_MAX_THREADS=0 - By default, this feature is disabled.
  - Defines the maximum number of threads (over all Fabric nodes) to be allocated.
- #ASYNC_LOAD_MAX_MEMORY_IN_MB=2147483648 - Default is 2GB
  - The maximum memory to be allocated for the parallel load process.

Re-design of Save Big LUI Chunks to Cassandra

Save Big LU chunks into Cassandra was re-designed to use the Cassandra Loader mechanism.

- The Loader configuration for the parallel save can be done per each LU by adding a new section named **<LU name>_ cassandra_entity_storage** to the config.ini. The parameters under this section are the same as Loader definition (for example, Loader execution mode).
- MAX_SAVE_MEMORY_USAGE was removed from config.ini.

Tomcat Upgrade

From 8.5.57 to 8.5.63.

RESOLVED ISSUES

- Ticket **21155**: In Fabric Studio, Database types => Studio Metadata => SQL Query for Columns List, the description is added to description of what is NULLABLE, as follows:
  - NULLABLE - Boolean true/false, String "true"/"false" or positive/negative number

- Ticket **21565**: In Fabric Studio, Web services => Serialize Null property, the issue of overwriting it from False back to True was fixed.
- Ticket **21584**: when a Broadway flow is updating a CommonDB table with no Transaction set on the stage, the following exception is now thrown: “Insert cannot be executed outside a transaction“.
- Several Broadway UI fixes related to collapsed stages, merged stages, redo / undo, actor’s input / output ports modification in the Actor Editor and other minor UI issues were fixed.
- Studio warning messages were improved.