



FABRIC V8.3.1 RELEASE NOTES

- These Release Notes describe the new features in Fabric release V8.3.1 and list bugs that have been fixed since the V8.3.0 release.
- Certification of this Fabric release is based on:
 - Cassandra version 4.1.3
 - SQLite version 3.46.1
 - OpenJDK Runtime Environment 21.0.4+7
 - Confluent Kafka version 8.1.1
 - Neo4j enterprise 5.26.18
 - Elasticsearch 8.5.3 and 7.16.3
 - PostgreSQL 17.5
 - AWS OpenSearch:
 - Cloud: 3.1 and 2.17
 - On prem: 1.3.4

MAIN FEATURES AND IMPROVEMENTS

1. Fabric Catalog

- **Transaction management while creating a graph in Neo4j:**
 - The Catalog version creation in Neo4j during the Discovery job has been improved. The identified changes are now split into batches and written to Neo4j as separate transactions, allowing the batches to run in parallel.
 - Splitting into batches significantly reduces memory consumption during Discovery job execution. It also significantly improves the efficiency and duration of graph creation, especially for very large source interfaces.
- https://support.k2view.com/Academy/articles/39_fabric_catalog/catalog_app/21_advanced_settings.html
- **Overriding Catalog generator input params by Catalog-calculated metrics:**
 - The existing Catalog plugins, Data Quality Metrics and Option Set Analyzer, calculate various data metrics – such as min, max, average, and the list of distinct values – and create the corresponding Catalog field properties for these metrics.
 - The values of these field properties can then be used by the masking and synthetic data generation processes to improve the quality of the generated values. To utilize them, an alias map needs to be created between the actor's input parameter and the related field property. The alias map enables overriding the actor's default input settings with the values of calculated metrics.



FABRIC RELEASE NOTES

- The alias map is established in the Advanced Masking Settings window, accessed via the Catalog Settings > PII & Masking tab.

https://support.k2view.com/Academy/articles/39_fabric_catalog/catalog_app/10_catalog_settings.html

https://support.k2view.com/Academy/articles/39_fabric_catalog/catalog_app/15_override_actor_input.html

- The **JsonSchemaToMetadata** Actor, used in the File Cataloging solution for parsing the JSON schema files, has been enhanced. It now provides full support for **Draft-07** and later versions of the JSON Schema specification.
 - The **JsonSchemaToMetadata** Actor transforms a JSON Schema document into a flat list of objects describing datasets, classes, and fields. It is performed by using the **NetworkNT Walk API** to traverse the schema, normalize references, and emit rows with enough metadata for downstream Catalog creation.
- The existing '**Option Set Analyzer**' plugin has been enhanced as follows:
 - It now assigns a new classification = **OPTION_SET** to each field, identified as an Option Set, unless that field has been already assigned a different classification.
 - The plugin can now operate in incremental mode that is set to 'Keep All' by default. Meaning that if the plugin has already been executed for a field in a previous Discovery Job execution, it will not run again for the same field.

https://support.k2view.com/Academy/articles/39_fabric_catalog/plugins/04_source_data_metrics.html

- A new '**Empty Datasets Discard**' plugin:
 - Some data sources may contain many empty tables that are irrelevant to Catalog and for further LU creation.
 - This plugin is introduced to improve Catalog usability as well as the LU development process, as it **excludes** all empty tables during Discovery.

https://support.k2view.com/Academy/articles/39_fabric_catalog/plugins/04_source_data_metrics.html

- **Catalog App UI** enhancements:
 - A new **Version Differences** window has been added to display a list of changes between two selected versions (per data platform and schema). The data displayed in the window can be exported to a CSV file. This window is available only in 'versions compare' mode.

https://support.k2view.com/Academy/articles/39_fabric_catalog/catalog_app/06_catalog_versioning.html

- When running **Build Artifacts from the Catalog App**, the **relations** artifact files (catalog_relations_info MTable) are also created in the Fabric Studio, in addition to the **field** artifact files (catalog_field_info MTable).

https://support.k2view.com/Academy/articles/39_fabric_catalog/catalog_app/09_build_artifacts.html



FABRIC RELEASE NOTES

- The **Crawler Filter** default settings in the **Discovery Pipeline** screen have been changed. Now, when adding a new rule, the filter is set by default to **Exclude Others** value, which suggests that the Crawler will filter out all entities except those written in the rule. The Crawler Filter can be changed to **Exclude This** or **No Filter**, if required.
- The **Data Snapshot** step of the Discovery monitor does not display the number of fields handled. It now displays only the percentage of step progress.
- Merging new classification generator settings into the project's MTable:
 - Fabric now automatically merges the existing project's PII & Masking settings (saved in catalog_classification_generators.csv and pii_profing.csv) with the product settings. This allows seamless merging of the product definitions into the project file.
 - In this version, a new classification, **OPTION_SET**, along with its generator, was added.
 - In V8.3.0, several new classifications, such as **DEFAULT_STRING** and other "defaults" were added along with their respective generator settings.
- Fixing the creation of the '*refers to*' relation properties in MySQL DB relations.
- Various Catalog App UI fixes.

2. Web Studio

- As part of Studio performance improvements, the setting 'Deploy all updated logical units on Studio start' is now changed to **false** by default. This means that Deploy should now be triggered manually when needed. This feature is especially helpful when working on projects that include a substantial large number of logical units.
- **Hide LUs** — enabling hiding LUs that the implementor is not currently working on. This allows all of the organization's assets to remain in the same source-controlled project while helping the implementor work more efficiently. When hidden, LUs do not appear in the Project tree and are excluded from the build and deploy process.

Hidden LUs are added to groups. This enables actions to be taken on either the group or a hidden LU. According to the settings, implementors can decide whether these hidden LU groups are preserved in source control or personalized for each implementor.
- **Graphit** — UX improvements:
 - An easy inclusion of another Graphit: When the user selects 'include' as the node type, a command builder action icon appears, and a pop-up window allows you to select the included Graphit.
 - Enabling the expansion of the results pane, making it easier to read it.
- **Environments** - Several changes were made to improve the user experience in projects with many environments and interfaces.



- Interface validation is now performed when opening an interface to improve loading time.
- Filtering: The list of environments in the upper horizontal pane can now be searched and filtered. Additionally, interfaces can now be filtered according to their status — active or inactive.

3. Miscellaneous

- **File System interfaces** attributes enhancements:
 - A **Recursive** indicator has been added to **all** File System interfaces (Local, SFTP, Azure, GCS and S3), to allow displaying all files in embedded folders.
 - **Files Filter Type** has been added to **Cloud** File System interfaces (Azure, GCS and S3), with two values:
 - **Wildcard** which supports filtering using *files wildcard pattern*.
 - **Regular expression** which supports filtering using *regex*.
- **Cloud File System interfaces** alignment:
 - The handling of all Cloud storage interfaces (Azure, GCS and S3) is now aligned with the Local File System interface.
 - Now, when running the **ls** command (or actor), the result transforms the “hierarchy” of the file name to match the local File System hierarchy.
 - Example: When file name in Azure is “lvl1/lvl2/file.txt”, the **ls** command will return “file.txt” (when Recursive is false).
 - For existing projects that want to preserve their current behavior:
 - Set `LEGACY_FILE_NAME = true` (in the relevant storage section of the config.ini file) to make the **ls** command return the full object key (path + name) instead of only the file name.
- Integration with Fortanix Data Security Manager KMS service — Fabric now supports storing its encrypted master key, which is used for data (LUIs) encryption in Fortanix KMS, which is a platform built on their Data Security Manager (DSM).
- When copying BLOBs from the Azure interface by using the MV actor, there used to be a hard 256MB size limit. This restriction has now been lifted, and the MV actor supports transferring large BLOBs.
- Node.js was upgraded to version 20.19.4.
- The LU Schema property ‘Enable data encryption’ is now selected by default for all newly created Logical Units, ensuring that data is encrypted by default.
- Fabric restart with watchdog:
 - If Fabric was started with a watchdog, restarting will kill the Fabric process while the watchdog remains running and ensures that Fabric starts again.



- OCI S3 (Oracle Cloud Infrastructure service compatible with the Amazon S3 API) is now supported by adding a new hidden parameter to the config.ini file, under the [s3_storage] section. The parameter is called CHECKSUM_VALIDATION (default value is True) and it controls S3 payload signing. Setting it to False disables the checksum validation.
- Caching Secret Managers secrets – Some secret manager providers restrict high-frequency access to secrets. To address this, Fabric can cache recently used secrets, reducing the number of requests to the secret manager. Configure the cache duration using the SECRETMANAGERS_CACHE_TTL parameter in config.ini. The default value is 60 seconds.
- The two legacy scripts stats-collection.sh and io-stats.sh were removed from the Fabric package.

RESOLVED ISSUES

- Ticket #43639 — switching between branches in Git did not work as expected.
- Ticket #44447 — the Catalog monitor did not correctly reflect the time spent on the 'Version Creation in GraphDB' step.
- Ticket #44855 — Web Studio project recompilation.
- Ticket #44522 — identifying an issue where Batch Job was not running at full capacity.
- Ticket #44945 — DbOne - Table Data Viewer failed when LU Name is not in lowercase.
- Ticket #45188 — fixing a limitation that prevented users from editing a classification within the PII & Masking tab.
- Ticket #45583 — Too many <defunct> processes in fabric server.
- Ticket #45678 — the FINAL GLOBAL in the subcategory can be updated via set_global global (despite documentation).
- Ticket #45705 — using FileWrite Actor to write to Local File System takes a long time when it is in transaction.
- Ticket #45758 — tables with more than 5k columns have had poor performance.
- Ticket #45772 — Catalog Discovery Pipeline UX.
- Ticket #45824 — resolving a Discovery job performance issue on Oracle SQL DB.
- Ticket #45917 — unable to stop a batch, which is using a Broadway flow with KAFKA subscriber as a source for the list of entities.
- Ticket #45919 — multiple zombie processes running in the environment.
- Ticket #46370 — jobs started to process after deployment, although they were scheduled to run at a specified time.
- Ticket #46700 — failure when linking Catalog Masking Mapper Actor with Concat Actor.
- Ticket #46674 — saving the Discovery Pipeline rules in the Catalog Settings has failed.



FABRIC RELEASE NOTES

- Ticket #46679 – Catalog ignored the *ConnectToSource = false* setting and threw an exception *'The connection attempt failed'*.
- Ticket #46752 – performance issue with LU Schema.
- Ticket #46753 – 'In process' entities are not visible on the admin batch GUI page.
- Ticket #45338 – environments file override – opening the WebStudio environments file editor resets search interface settings from Secret Manager to defaults (related to old ticket #42099).
- Ticket #45505 – JWT can fail when it becomes too large, for example if the SSO provider includes many group claims for the connected user.
- Ticket #45996 - Customized APIs can now also be set and used with the PATCH verb, either when using Graphit or Java web-services.
- Ticket #45676 - Web Studio > Environments, the capability to edit “final” Globals is now disabled.
- Ticket #46326 – Unzip Actor is now working properly in debug mode.
- Reference deployment activity is not always updating the globals values defined in the shared level.
- Removed an extra & character that was incorrectly added to the URL used to call CyberArk as the secret manager.