DB Gene 4.5.0 Release Notes

March 11th, 2024

DB Gene 4.5.0 Release Notes

Note:

DB Gene 4.5.0 introduces several infrastructure updates and deprecations.
They are described in DB Gene 4.5.0 Migration Guide, available on the DecisionBrain website.

The following information only focuses on the main changes in this release.

Important Notes

*Please take into account the following important information when using the new version of the **DB Gene 4.5.0**, released on March 24th, 2025.*

Updates

DB Gene 4.5.0 introduces the following infrastructure updates.

Spring Update

DB Gene 4.5.0 now uses **Spring Boot 3.4.3** and **Spring Cloud 2024.0.0**. Formerly, it was version 3.3.5 and version 2023.0.3.

Angular Update

DB Gene 4.5.0 now uses **Angular 19.1.7**. Formerly, it was version 18.2.10.

AG Grid Update

DB Gene 4.5.0 now uses **AG Grid 33.1.1**. Formerly, it was version 31.2.0.

Keycloak Update

DB Gene 4.5.0 now uses **Keycloak 26.1.3**. Formerly, it was version 26.0.5.

MongoDB Update

DB Gene 4.5.0 now uses **MongoDB 8.0.5**. Formerly, it was version 6.0.5.

PostgreSQL Update

DB Gene 4.5.0 now uses **PostgreSQL 15.10**. Formerly, it was version 15.5.

RabbitMQ Update

DB Gene 4.5.0 now uses **RabbitMQ 4.0.7**. Formerly, it was version 4.0.2.

Deprecations

- In 4.5.0, the Custom Dashboards layout can be configured with a "fit to screen" option, which allows the positioning of a single widget to fit the whole screen just like in a Custom View.

  Therefore, Custom Views are now deprecated and will be removed in the next version.

- In in `GeneWidgetManifest`:
  - `minItemCols` is now deprecated and will not be taken into account on a dashboard, all widgets can be resized to take up only one column.
  - `minItemRows` is now deprecated and will not be taken into account on a dashboard, all widgets can be resized to take up only one row.

- The task `SimpleExcelExportTask` has been removed.

- The `ExcelExportRoutine` routine has been deprecated in 4.0.1-fp2 and will be removed in the next version.

- In the Data Service, the endpoints `api/data/simple-excel-export` and `api/data/simple-excel-import` have been removed.

- The method `show_api_key_input(api_key)` from the class `GeneHelper` of the DB Gene jupyter library has been deprecated in favor of `set_api_key(api_key)`.
- The method `show_scenario_picker(parameter_name)` from the class `GeneHelper` of the DB Gene jupyter library has been deprecated in favor of `select_scenario(collector_class, parameter_name)`.

End-User Features

DB Gene 4.5.0 introduces changes to the JupyterLab integration, through improved APIs and samples, and now provides virtual environments to install third-party libraries. This version also improves the UI, especially on dashboards. New options are available for the Chart, Gantt Chart, Code Editor, Filter widgets, and Pivot Table. Finally, Gene Online has been notably enhanced and now provides its users with AI-powered JDL parsing.

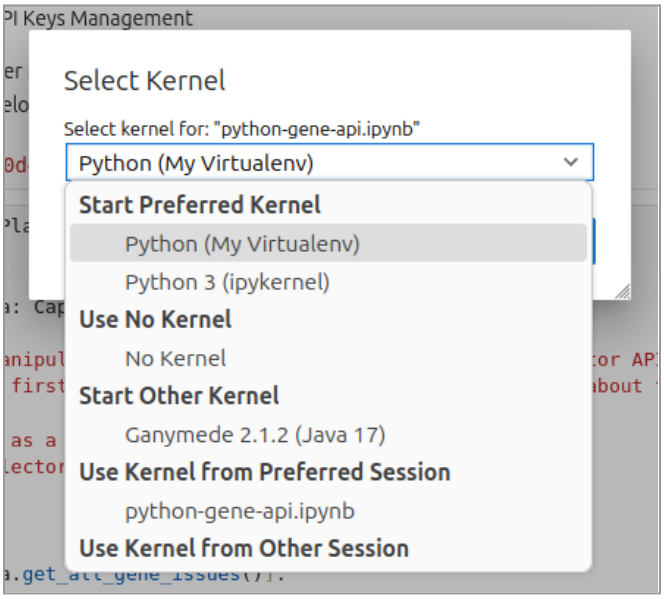
Improved JupyterLab integration

JupyterLab integration has been improved with the following improvements.

Improved JupyterLab Compatibility with 3rd-Party Libraries

Jupyter Notebook can now use 3rd party libraries.

When starting JupyterLab for the first time, you see in the log that the virtual environment is created. When using Kubernetes, the log indicates that the sample notebooks and requirements have been copied



After restarting JupyterLab, the new virtual environment kernel should still be available in the JupyterLab kernel list.

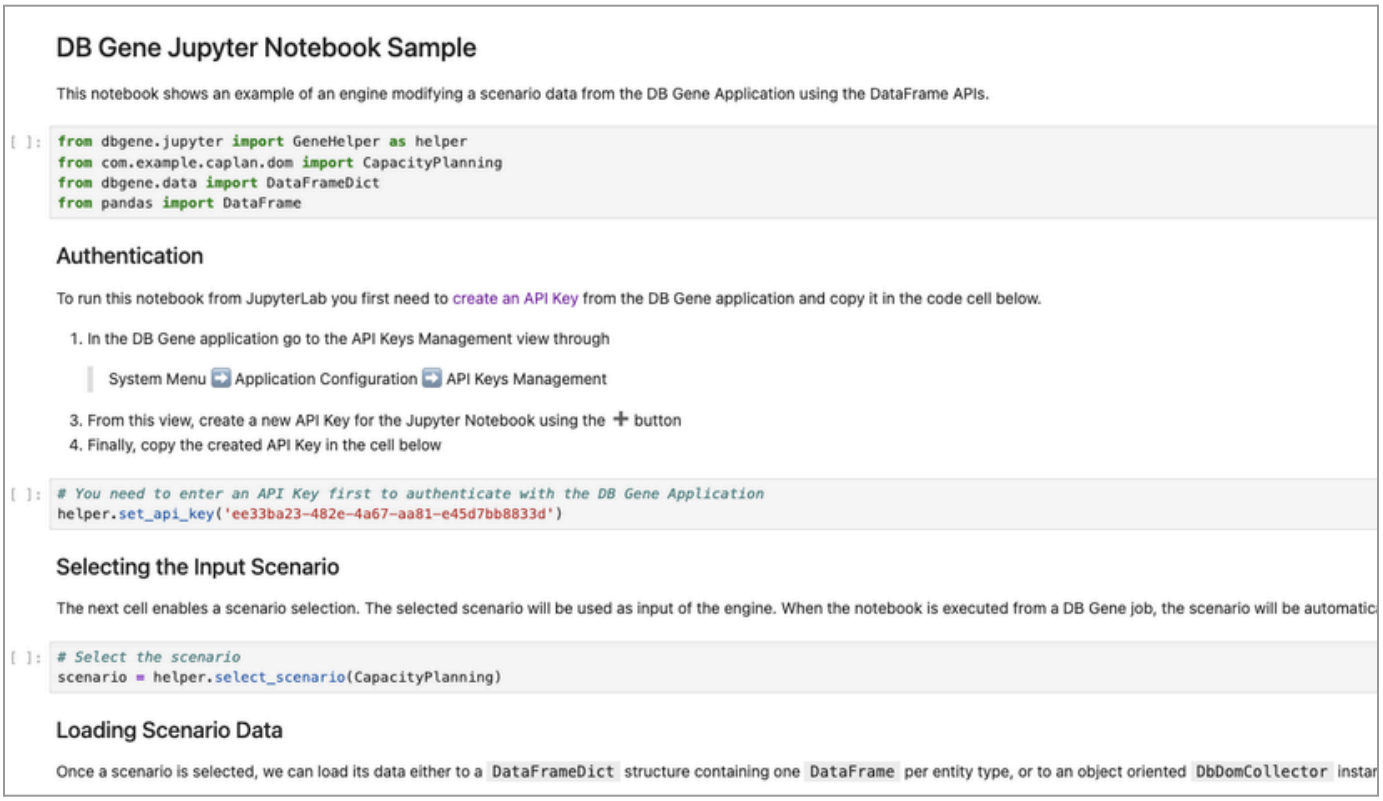
Installing a Python module in the virtual environment kernel is possible by adding it to the file `requirements.txt` and running the command `virtualenv-kernel`.

When using Gene Online, changing the data model restarts JupyterLab with the new model available for Python

Improved Jupyter Notebook Samples

A new notebook, `gene_notebook.ipynb`, is available in the notebook list and is a simplified version of the previous one, showing how to load/save and modify data using dataframes.

In addition, the sample code is split into more cells, each documented at the top of the notebook, to display information about its execution.



DB Gene Jupyter Notebook Sample

This notebook shows an example of an engine modifying a scenario data from the DB Gene Application using the DataFrame APIs.

```
[ ]: from dbgene.jupyter import GeneHelper as helper
     from com.example.caplan.dom import CapacityPlanning
     from dbgene.data import DataFrameDict
     from pandas import DataFrame
```

Authentication

To run this notebook from JupyterLab you first need to create an API Key from the DB Gene application and copy it in the code cell below.

1. In the DB Gene application go to the API Keys Management view through

   System Menu ➡ Application Configuration ➡ API Keys Management

3. From this view, create a new API Key for the Jupyter Notebook using the + button
4. Finally, copy the created API Key in the cell below

```
[ ]: # You need to enter an API Key first to authenticate with the DB Gene Application
     helper.set_api_key('ee33ba23-482e-4a67-aa81-e45d7bb8833d')
```

Selecting the Input Scenario

The next cell enables a scenario selection. The selected scenario will be used as input of the engine. When the notebook is executed from a DB Gene job, the scenario will be automatic

```
[ ]: # Select the scenario
     scenario = helper.select_scenario(CapacityPlanning)
```

Loading Scenario Data

Once a scenario is selected, we can load its data either to a `DataFrameDict` structure containing one `DataFrame` per entity type, or to an object oriented `DbDomCollector` instar

New Jupyter Notebook Task and Statement

A JupyterLab notebook can now be executed from the application using:

- a new task "*Execute a Jupyter Notebook*" and
- a new statement `ExecuteJupyterNotebookStatement`.

Updated JupyterLab Helper

The JupyterLab Helper has been improved and now uses XCSV instead of CSV API.

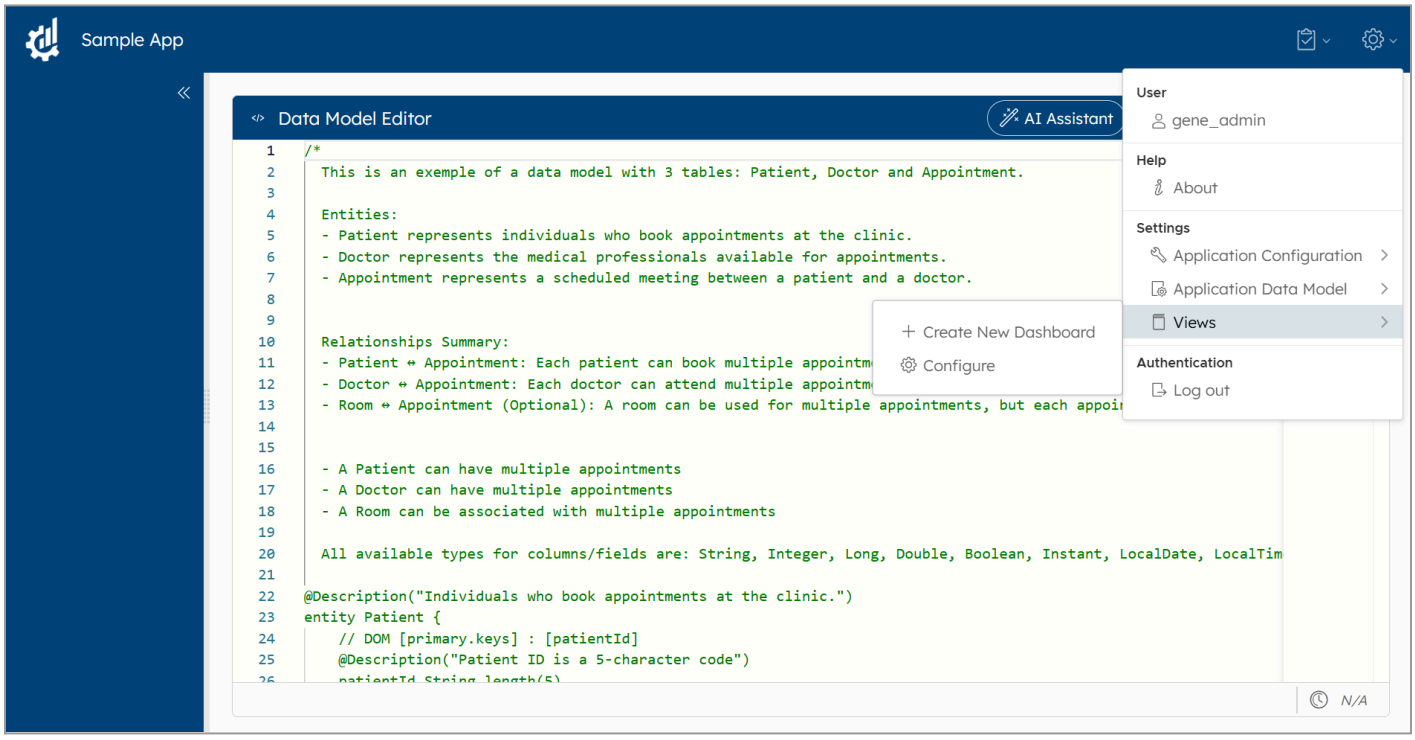
Improved Gene Online

Gene Online allows developers to deliver applications faster with the following improvements.

Improved Default Configuration

By default in Gene Online:

- The notions of "Views" and "Workspaces" are now disabled. Therefore, the Workspace Selector and the "Views" separator have been removed from the Sidenav.
- Also in the Sidenav,
    - The button *"Home"* is now hidden by default and can be restored by setting *SIDEBAR_SHOW_HOME* to true in the Application Preferences. The Home view is still available by clicking the application name next to the logo.
    - A button *"+ New Dashboard"* is available.
- The default Filter Scope is now set to "VIEW".
- A blank JDL Data Model Editor now shows a commented example. The `application` block support has been removed and is now ignored.



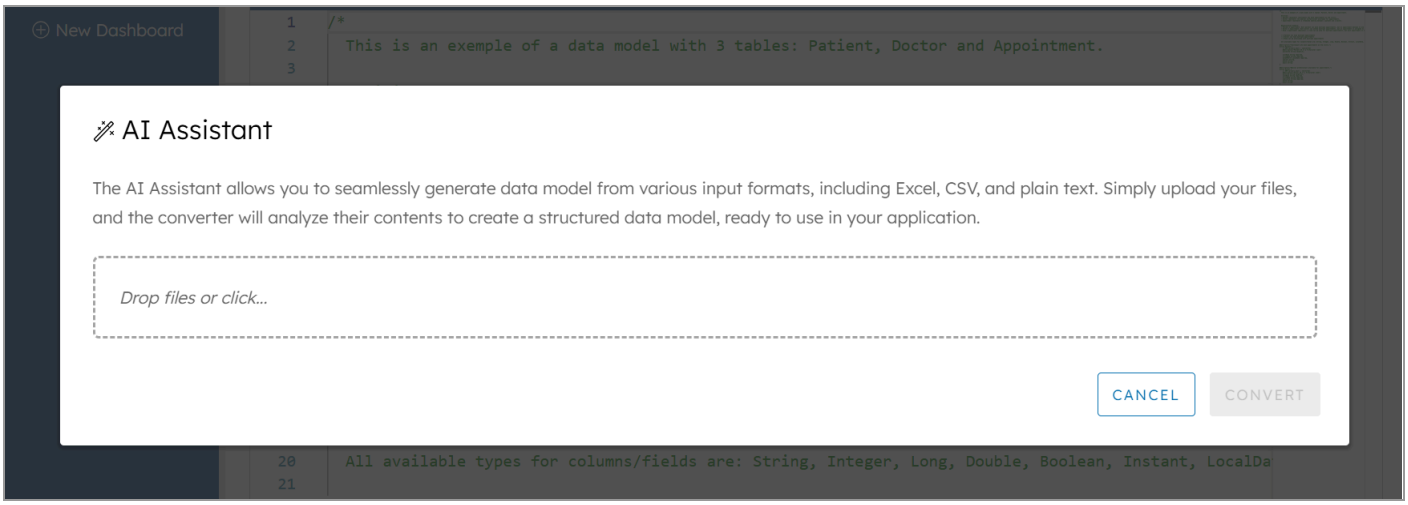
New JDL AI Assistant

A JDL AI Assistant has been integrated. It allows the generation of a data model from different types of files, such as Excel, text, image, or audio files.

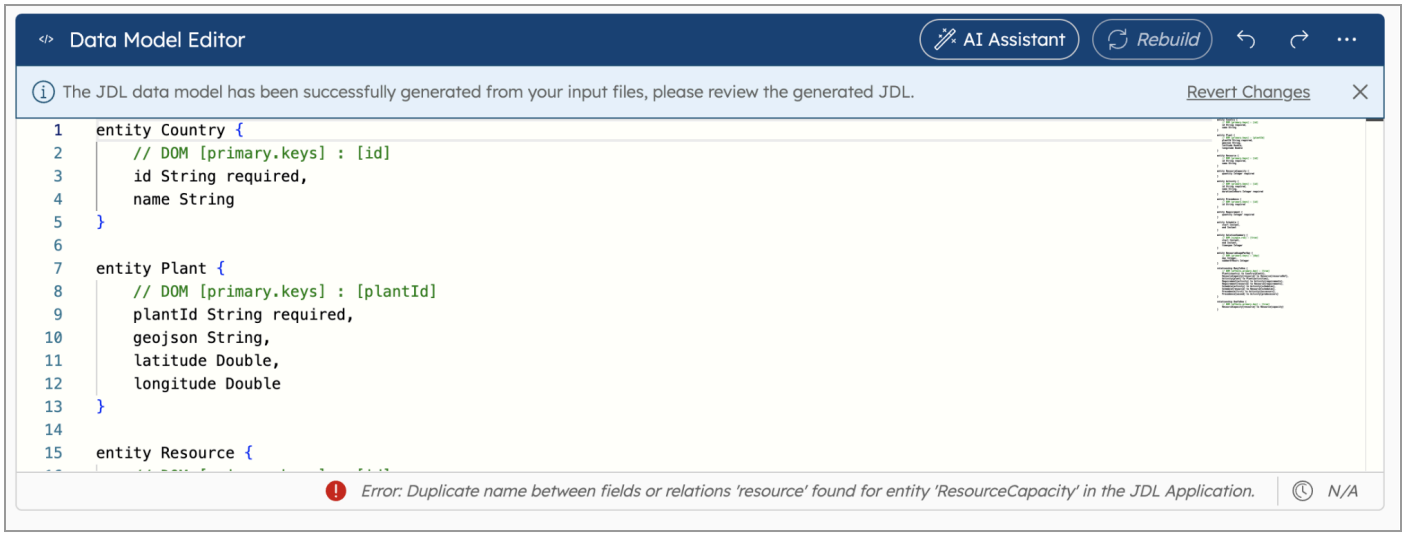
```
/*
 This is an exemple of a data model with 3 tables: Patient, Doctor and Appointment.

 Entities:
 - Patient represents individuals who book appointments at the clinic.
 - Doctor represents the medical professionals available for appointments.
 - Appointment represents a scheduled meeting between a patient and a doctor.


 Relationships Summary:
 - Patient ↔ Appointment: Each patient can book multiple appointments, but an appointment belongs to one patient.
 - Doctor ↔ Appointment: Each doctor can attend multiple appointments, but an appointment is assigned to one doctor.
 - Room ↔ Appointment (Optional): A room can be used for multiple appointments, but each appointment is held in one room.
```



The AI assistant is available in the JDL Data Model Editor toolbar.



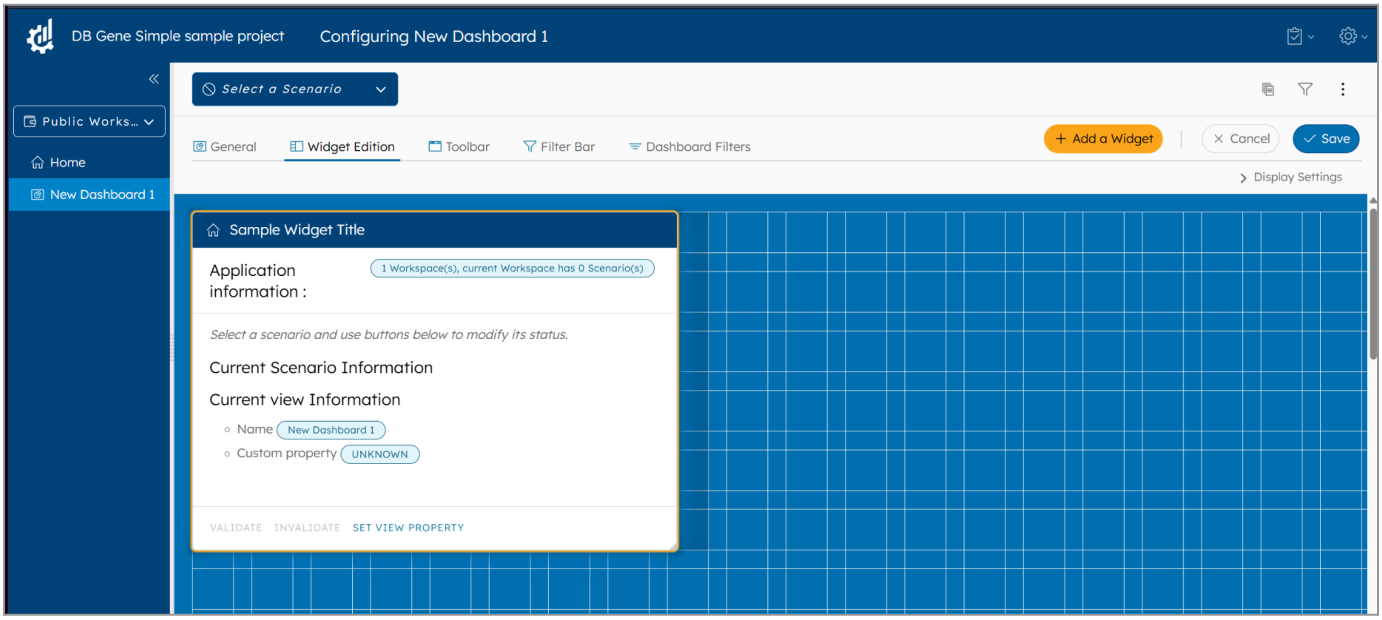
Improved Application Display

Version 4.5.0 introduces several improvements in terms of application display.

Improved Dashboard Configuration

The items "*Configure Dashboard*" and "*Edit Layout*", formerly available from the toolbar menu ⋮, are no longer available. Instead, the button "*Edit*" on the toolbar opens the dashboard configurator.

The "Widget Layout" can be configured in a dedicated tab with a new collapsible section to edit specific global settings.



In addition:

- In the tab "*Toolbar*", a new option allows showing or hiding the toolbar on the dashboard.
- In the tab "*Filter Bar*", a new option allows setting the default visibility of the Filter Bar to:
  - "*Expanded*"
  - "*Collapsed*"
  - "*Use Preferences Settings*" is controlled by the Application Preferences setting, *DEFAULT_FILTER_BAR_STATE*. By default, it is set to *open*. When this setting is *closed*, the Filter Bar of all views/dashboards with the default visibility set to "*Use Preferences Settings*" will be collapsed by default.

Improved Sidenav and Topbar Menus

The colors of the Sidenav bar now reflect the application colors. The legacy Sidenav colors can be restored by setting *USE_LEGACY_SIDEBAR_COLORS* to *true* in the Application Preferences.

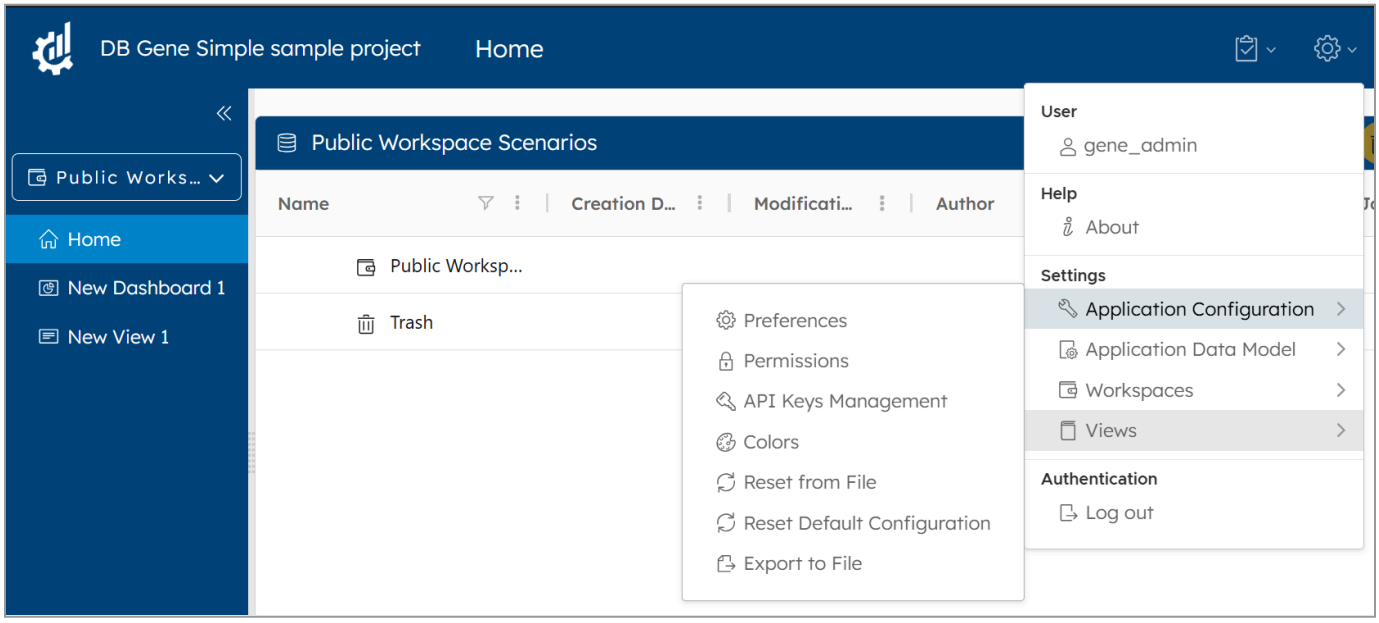
Also, as the notion of "views" and "dashboards" is being reworked, the *Views* separator has been removed from the Sidenav.

In the Topbar Settings menu:

- "*Views and Dashboards*" has been renamed "*Views*".
- "*Application Preferences*" is now named "*Preferences*", and it has been moved under "*Application Configuration*".

Finally, on the left side of the Topbar:

- Clicking the application name now redirects to the Home view, and
- The name of the view or dashboard is not underlined when hovered anymore, as it could be confused for a link.



Improved Application Preferences

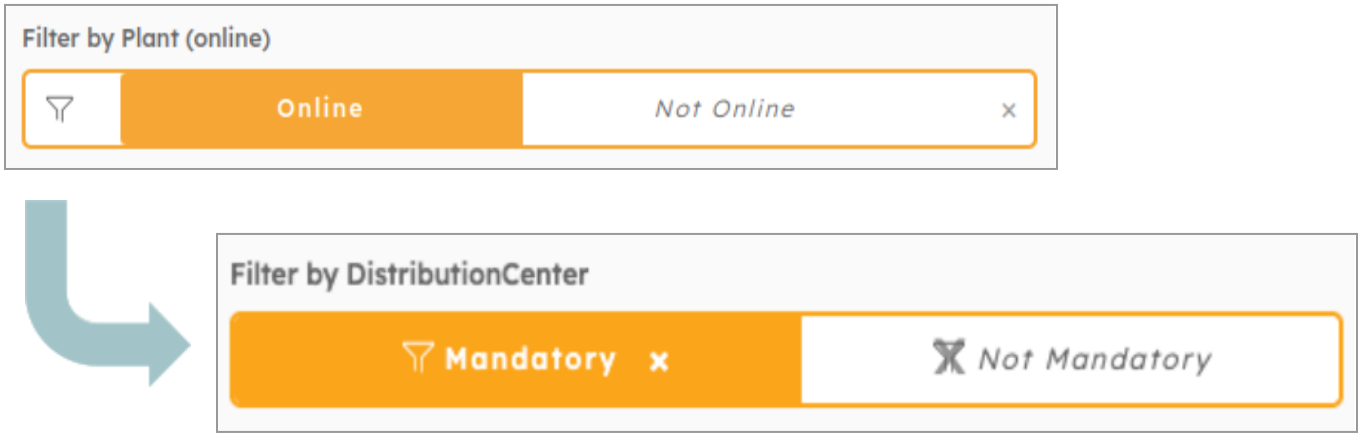
New Application Preferences settings are available:

- When set to *true*, *SHOW_GENE_DASHBOARD_LAYER_BUTTON* allows restoring the former menu to order widget layers in edition mode. The default is *false*.
- When set to *true*, *USE_LEGACY_SIDEBAR_COLORS* allows restoring the former colors for the Sidenav. The default is *false*.
- When set to *false*, *TASK_MENU_SHOW_JUPYTERLAB* allows hiding the entry "*JupyterLab*" in the Topbar Tasks menu. The default is *true*.
- When set to *closed*, *DEFAULT_FILTER_BAR_STATE* allows hiding the Filter Bar of all views/dashboards configured with the option "*Use preference setting*". The default is *open*.

Improved Boolean Filter Display

For boolean filters:

- The icon is now on each button, crossed out on the "false" option, instead of on the left-hand side.
- The clearing button is now on the active button, instead of on the right-hand side.
- The buttons take up the whole filter space.
- Inactive filter styling options are added when no scenario is selected or the filter does not apply to the current scenario in a CDM application.

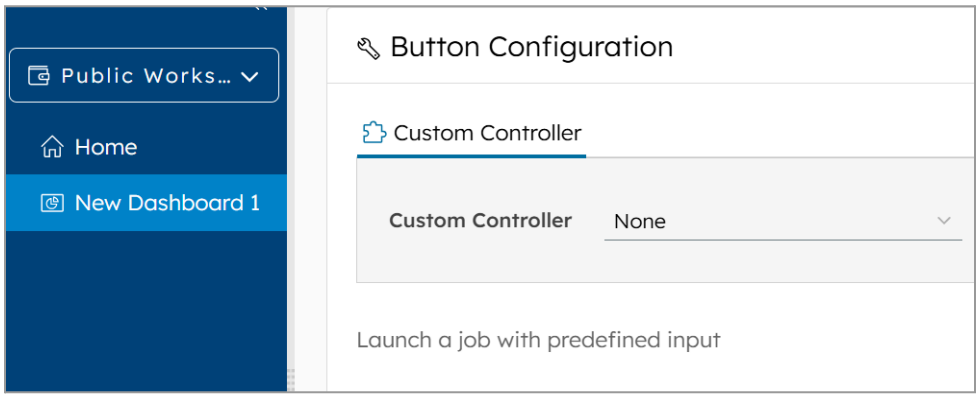


Improved Widgets

This version introduces several improvements regarding widgets.

Improved Widget Configuration

The widget type and description are now displayed in the widget configurator.



Improved Gantt Chart Widget

Version 4.5.0 introduces several improvements in the Gantt Chart widget:

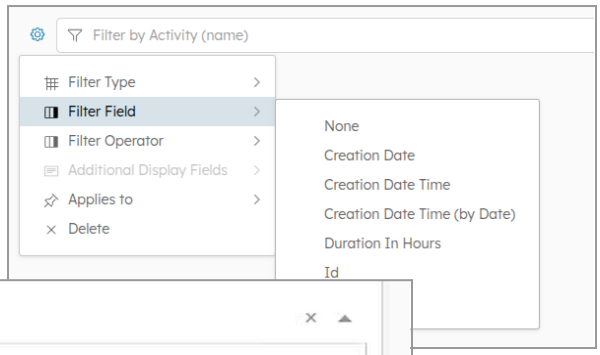
- Data edition now supports snap to grid, drag and drop scrolling, and deleting events.
- The Custom Controller API now supports controlling editions.
- The Gantt Chart API now supports rendering images

Finally, the `readUpdates` method of the `GanttDataSource` interface now returns a boolean to indicate if updates have been applied. This method should return `true` and request a refresh on the Gantt chart itself in this case. If it returns `false`, the Gantt component may refresh after calling `readUpdates`.

Improved Filter Widget

The option "Applies to" can now be edited when a "Filter field" is selected (on the right).

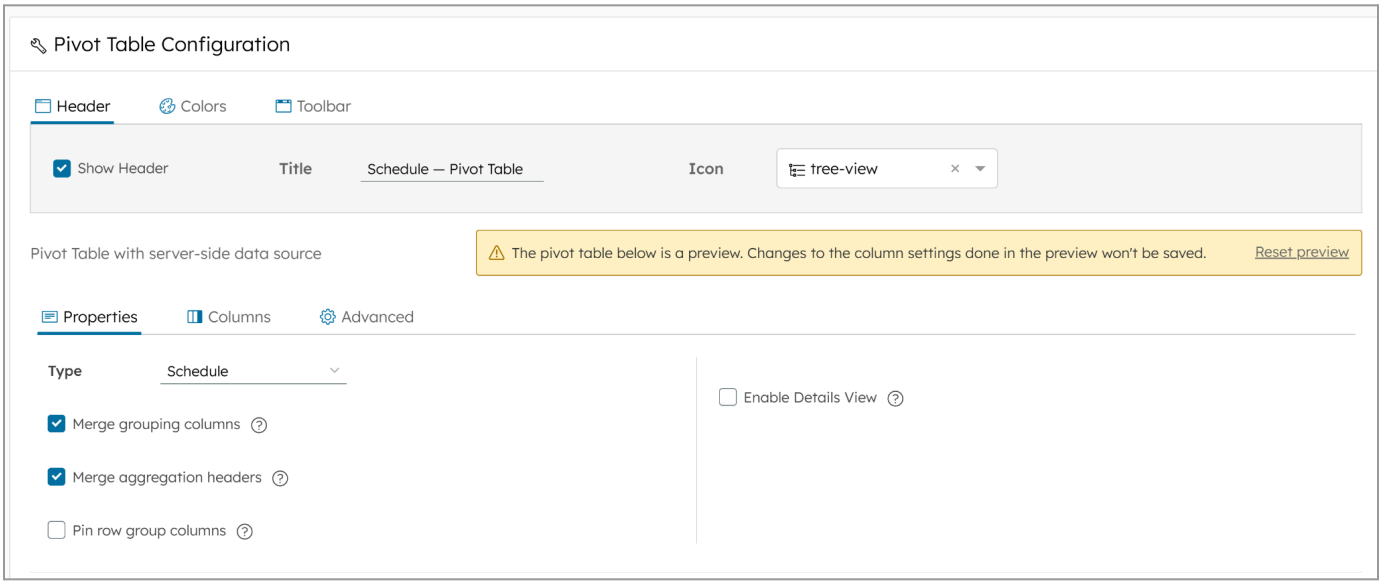
In addition, the Filter widget labels used in display and autocompletion can now be configured (below).



Improved Pivot Table Widget

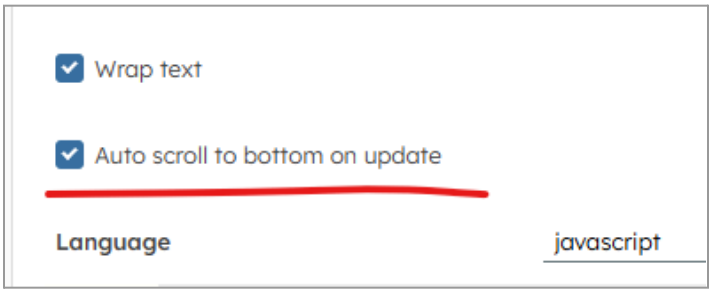
The Pivot Table widget configurator now provides users with new options:

- Pin row groups
- Auto expand on scroll
- Aggregated Rows Views
  - Double-clicking on a cell opens a table with all the aggregation lines.
  - If the column is editable, you can edit the value.
  - When saving, the pivot table is refreshed



Also, the data edition pop-up table can now be displayed next to the pivot table.

Finally, the Server Pivot Table allows users to freeze the columns enabled in the "*Row Groups*" section. This feature will ensure these columns remain visible while the user scrolls horizontally.



Improved Code Editor Widget

A new "Auto scroll to bottom on update" option in the Code Editor widget configurator is enabled by default.

Improved Data Handling

This version improves the data model and scenario data handling, especially for Python.

Improved JDL Validation

JDL validation of the data model now returns all syntax errors at once.

Improved Python DOM API

The Python DOM API now supports the following methods:

- `__str__()`, allowing user-friendly string rendering of DOM objects.
- `display(entity)`, allowing user-friendly HTML rendering in Jupiter notebooks. For more details on defining custom display logic, refer to the Jupyter Notebook Community.
- `get_all_xxxx()`, as in `input_collector.get_all_issues()`, and `remove_all_xxxx()`, as in `input_collector.remove_all_issues()`, to get and remove all instances of an object.

Also, when reading a CSV from Python, "null" values are kept as `null` instead of an empty string.

Improved Data API

Data API no longer exposes “internal IDs” of relations in Python dataframes.

Improved Export

Users with ACCESS-only permission can now export scenarios. Also, “Internal IDs” are now optional in XCSV export.

Improved Tasks

Improved and new variables, methods, and statements are now available for tasks and routines.

Improved Exit Script in Tasks

The exit script of a scripted task now has access to the variable named `com.decisionbrain.gene.latestStepLabel`, similar to the variables that contain the job status and error message from the main script, if any.

This new variable contains:

- `null`, if the main script ends normally, or
- The label of the step that was interrupted, as seen in the Job Details widget.

Improved Method in Composite Data Model Routines

When using routines in a Composite Data Model application, the `markModified` method now applies to all visible reference scenarios.

New Statement to Execute a JupyterLab Notebook

The statement `ExecuteJupyterNotebookStatement` has been introduced and documented. It executes the cells of a JupyterLab notebook on a given scenario.

The scripted task `ExecuteJupyterNotebookTask` has also been introduced. It is activated with the `@AddJupyterLabTask` annotation. It allows calling the above statement and can typically be executed from an action set on a toolbar button.

The `ExecuteJupyterNotebookTask` uses an API key instead of a JWT token to load and save scenarios.

New Built-in Routine for Rule Execution

The routine `ExecuteRulesetOnScenario` has been introduced and documented. It compiles and executes a set of rules written in Drools Rule Language on the data of a scenario. It is activated by adding the `@AddExecuteRulesetOnScenarioRoutine` annotation in the configuration of the Backend service.

Scripted task `ExecuteRulesetOnScenarioTask` has also been introduced. It is activated with the `@AddExecuteRulesetOnScenarioTask` annotation. It allows calling the above routine and can typically be executed from an action set on a button of the Code Editor widget toolbar. It is used by the Rules Script Editor widget.

New Built-in Routine for ChatGPT Interaction

The routine `ProcessChatGptConversationRoutine` has been introduced and documented. It processes and extends a conversation with OpenAI's GPT large language models. It is activated by adding the `@AddChatGptConversationRoutine` annotation in the configuration of the Backend service.

The scripted task `ProcessChatGptConversationTask` has also been introduced. It is activated with the `@AddChatGptConversationTask` annotation. It allows calling the above routine and can typically be executed from an action set on a button of the Code Editor widget toolbar.

Technical Features

DB Gene 4.5.0 introduces several technical improvements regarding DBOS, job memory usage, Gradle, JupyterLab, and deployment.

Improved DBOS

All the CRON in the DBOS scheduler are now configurable via an environment variable.

In addition, DBOS worker KPIs are now displayed in the DBOS console.

Improved Job Memory Usage

The memory needed to save the task outputs for a list of files is now reduced.

Improved JupyterLab Processing Files

The implementation of JupyterLab has been changed. Almost all files in `processing/jupyterlab/*` have been reworked.

Pay attention to the change in the default kernel name of the scaffolded Python notebooks.

It is necessary to update the following section in any existing Python notebooks:

```
Unset
    "kernelspec": {
     "display_name": "Python (default-virtualenv)",
     "language": "python",
     "name": "default-virtualenv"
    }
```

Improved Deployment Environment

Keycloak deployments (Helm and Docker Compose), the following deprecated environment variables have been migrated to the new ones:

- Updated `KEYCLOAK_ADMIN` for `KC_BOOTSTRAP_ADMIN_USERNAME`.
- Updated `KEYCLOAK_ADMIN_PASSWORD` for `KC_BOOTSTRAP_ADMIN_PASSWORD`.

According to the Keycloak migration guide, the following environment variables have been removed:

- `KC_HOSTNAME_STRICT_HTTPS=false`
- `KC_PROXY=edge`

And the following one is introduced:

- `KC_HTTP_ENABLED=true`

Changelog

DB Gene 4.5.0 introduces several improvements and bugfixes listed in detail below.

Improvements

DB Gene 4.5.0 introduces the following improvements:

Application *Views & Dashboards*	*DOC-1027*	*Dashboards/Views now have an option to set the Filter Bar default visibility*
	DBPF-7829	*Widget manifest attributes minItemCols and minItemRows are now deprecated and will no longer be taken into account*
	DBPF-7830	*Dashboard margin settings are now more streamlined*
Data *Built-in Import/Export*	*DBPF-7563*	*The SimpleExcelExportTask task has been removed*
	DOC-1047	*Users with ACCESS only permission can now export scenarios*
	DBPF-7755	*Internal IDs are now optional in an XCSV*
Data *Database*	*DBPF-7842*	*Data API no longer exposes internal IDs of relations in Python Dataframes*
Data *Generation*	*DOC-1079*	*Python DOM generated code is using reserved keyword '"yield"'*
Data *JDL*	*DBPF-7745*	*JDL validation now returns all syntax errors at once*
Data *Model*	*DBPF-5626*	*Python DOM - A new remove_all_xxx() API is now available*

DBOS *Master*	*DBPF-5531*	*All DBOS CRON are now configurable*
Dev *3rd-party Components*	*DBPF-7182*	*Trivy - Postgres was updated to version 15.10*
	DBPF-7674	*Updated Spring Boot from 3.3.5 to 3.4.3*
	DBPF-7674	*Updated Spring Cloud from 2023.0.3 to 2024.0.0*
	DBPF-7674	*Updated Spring Framework from 6.1.14 to 6.2.3*
	DBPF-7674	*Updated graphql-java-extended-scalarsto version 22.0*
	DBPF-7674	*Updated Apache Commons Compress from 1.26.1 to 1.27.1*
	DBPF-7674	*Updated Apache Commons IO for 2.16.1*
	DBPF-7674	*Updated Open API Generator from 7.9.0 to 7.12.0*
	DBPF-7678	*Updated Angular from 18.2.10 to 19.1.7*
	DBPF-7681	*Updated ag-grid-angular from version 31.2.0 to version 33.1.1*
	DBPF-7681	*Updated ag-grid-community from version 31.2.0 to version 33.1.1*
	DBPF-7681	*Updated ag-grid-enterprise from version 31.2.0 to version*

	DBPF-7263	Updated Keycloak from 26.0.5 to 26.1.3
	DBPF-7676	Updated MongoDB from 6.0.5 to 8.0.5
	DBPF-7182	Updated PostgreSQL to version 15.10 (alpine version)
	DBPF-7677	Updated RabbitMQ from 4.0.2 to 4.0.7
	DBPF-7188	Updated Java service to latest JDK 17
	DBPF-7197	Updated Bouncy Castle libraries from 1.70 to 1.78.1
	DBPF-7193	DBOS - Updated Tools to python:3.12.8-alpine3.21
	DBPF-7195	DBOS - Updated to Java-WebSocket >= 1.6.0
	DBPF-7190	DBOS - Updated NGINX has been updated to 1.27.3 in DOC Web base image in UI and documentation
	DBPF-7191	DBOS - Updated Master to latest Ubuntu Noble base image noble-20240904.1
	DBPF-7198	DBOS - Updated to springdoc-openapi-starter-webmvc-ui 2.0.3
	DBPF-7199	DBOS - Updated kubernetes to client-java >= 20.0.0
	DBPF-7203	DBOS - Updated jszip from version 3.7.0 to version 3.10.1.
	DBPF-7204	DBOS - Updated luxon from version 2.3.2 to version 3.5.0.
	DBPF-7205	DBOS - Updated groovy-all to version 4.0.24
	DBPF-7205	DBOS - Updated springdoc-openapi-starter-webmvc-ui to 2.6.0
	DBPF-7205	DBOS - Updated springdoc-openapi-starter-common to 2.6.0
	DBPF-7205	DBOS - Updated mongock has been updated to 5.5.0
	DBPF-7205	DBOS - Updated swagger-ui webjar to >= v4.18.2
	DBPF-7206	DBOS - Updated to testng >= 7.5.133.1.1

	DBPF-7681	DBOS - Updated ag-grid-angular from version 31.2.0 to version 32.3.3
	DBPF-7681	DBOS - Updated ag-grid-community from version 31.2.0 to version 32.3.3
	DBPF-7681	DBOS - Updated ag-grid-enterprise from version 31.2.0 to version 32.3.3
	DBPF-7201	Java workers - Updated aws-java-sdk-s3 to 1.12.780
	DBPF-7232	DOC Generator (Scaffolder) - Updated base Docker image to nodejs-20-minimal:1-1736729465.
Dev Gene Online	DBPF-7489	Application block is now optional in the GeneOnline JDL editor
	DBPF-7528	The default Filter Scope is now set to VIEW for Gene Online
	DBPF-7531	"Home" is now hidden from the Sidebar for Gene Online
	DBPF-7532	The "Workspace" notion is now hidden for Gene Online
	DBPF-7534	"Views" notion is now hidden for Gene Online
	DOC-382	The "Views" separator has been removed from the sidebar
	DBPF-7538	The colors of the sidebar reflect the application color for Gene Online
	DBPF-7546	The Dashboard Configuration and Layout menus have now been combined into one configuration view
	DBPF-7547	The toolbar configuration now has a new global settings section to show/hide the toolbar on the dashboard
	DBPF-7548	The Widget layout view has a new collapsible section to edit Gridster display settings
	DBPF-7550	UX research results to improve the widget display in layout mode are now applied
	DBPF-7555	The type of widget is now displayed in the widget's configuration view
	DBPF-7556	The Widget Configuration now contains the widget description

	DBPF-7638	Python DOM API now implements __str__() and display methods
	DBPF-7639	Python DOM API has been improved
	DBPF-7643	Blank JDL now shows a commented example with no application block
	DBPF-7648	Jupyter Notebook can now use 3rd party libraries
	DBPF-7708	GeneOnline now has a JDL AI Converter
Dev JupyterLab	DBPF-7409	A simplified Jupyter Notebook sample is now available
	DBPF-7475	A JupyterLab notebook can now be executed from a DOC Application
	DBPF-7636	Jupyterlab helper now uses XCSV instead of CSV API
	DBPF-7709	Jupyter Notebook Samples have been improved
	DBPF-7710	Jupyter Notebook Gene Helper have been improved
Dev Python	DBPF-5454	Python — load CSV data now differentiating null and empty strings
	DBPF-7271	[Tech Debt] Change the Python CSV Dialect to quoting csv.QUOTE_NOTNULL
Scripted Tasks Definition	DBPF-7487	[Technical Dept] ExecuteJupyterNotebookTask use Api Key instead of JWT token for loading and saving scenarios.
	DOC-1086	The exit script of a task can now know which step failed during the main script execution
Scripted Tasks Jobs	DBPF-7595	The memory usage for List<FileValue> is now reduced
Scripted Tasks Routines	DBPF-7127	markModified now applies to visible scenarios as well

UI General Changes	DBPF-7337	The new Boolean filter display is now improved
UI Application Preferences	DBPF-7535	Application Preferences is now placed in the Application Configuration section
UI Chart	DOC-809	[Chart] Y-axis labels are now left-aligned with the axis
UI Code Editor	DOC-1017	Add 'auto-scroll-to-bottom' setting in Code Editor
UI Filter	DBPF-7219	[UI - Filter] "Applies to" can be edited when a "Filter field" is selected
	DOC-948	Display Descriptive Values Alongside Keys in Filter Widgets
UI Gantt	DBPF-7240	Edition Mode
	DBPF-7245	Gantt Custom Controller API now supports controlling edition
	DBPF-7246	[Gantt] Gantt data edition now supports snap to Grid
	DBPF-7574	[Gantt] Gantt edition now supports deleting events
	DBPF-7612	[Gantt] Chart drawing API now supports rendering images
	DBPF-7631	[Gantt] Drag'n'drop scrolling is now supported in edition mode
UI Pivot Table	DBPF-7328	The data edition pop-up table can now be displayed next to the pivot table
	DOC-940	Pivot table now has new functionalities

Bugfixes

DOC 4.5.0 introduces the following bugfixes:

Application *Views & Dashboards*	*DBPF-7739*	*View toolbar is shown on Home view if coming from a dashboard that shows it*
	DBPF-7746	*Widgets from old configurations were not draggable anymore in Layout mode*
	DBPF-7845	*There were some bugs with the new Dashboard Edition view*
Data *Built-in Import/Export*	*DOC-1065*	*The Excel Template could omit the three Gene... tabs*
Data *Database*	*DBPF-7836*	*Boolean are not saved from a Python DataFrame*
Data *Model*	*DOC-1012*	*idOfNewScenario is failing with a correct error message for CDM*
Data *Scenario Service*	*DOC-1033*	*SQL Table name resolution could return a corrupted name when multiple scenarios were created in parallel*
	DBPF-7705	*Loading a python scenario collector fails with an access denied*
DBOS *Master*	*DOC-989*	*Expired JWT token*
	DOC-1007	*DBOS job fails with ""Worker has died"" after ""worker stopped"" when worker terminated normally*
	DOC-1070	*DBOS master - Out Of Memory error when worker write output*
DBOS *Web Console*	*DOC-1003*	*[DBOS console] Series in the KPI chart could not be unselected anymore*

Dev *Build*	*DBPF-7579*	*Repository URLs were not updated automatically in the gradle.properties*
Dev *CPLEX*	*DBPF-7780*	*There were issues on CPLEX IloMemoryException with Native Jar*
Dev *Gene Online*	*DBPF-7559*	*GeneOnline — JupyterLab rebuild and restart were sometimes failing*
	DBPF-7645	*JDLs with unclosed comment blocks were considered valid in Gene Online*
	DBPF-7692	*The helper API for JupyterLab notebooks has been reworked*
	DBPF-7725	*Rule Scripts Editor couldn't be added to a composite widget*
	DBPF-7731	*Java DOM code generator couldn't compile generated classes*
Dev *Deployment*	*DOC-1055*	*updateCode/updateCodeCheck does not update the versions in the helm chart*
Dev *Security*	*DOC-927*	*Gene Actuator could return NaN gene_idle_time_seconds under some conditions*
	DBPF-7592	*NPE was triggered when executing a Ruleset*
	DOC-1044	*Keycloak Realm export does not work anymore*

Scripted Tasks *Jobs*	*DOC-1014*	*Starting the scenario creation task from ""Run new job"" was not loading the references*
Scripted Tasks *Routines*	*DOC-922*	*Routines were crashing when returning a list of files*
UI *Button*	*DOC-1023*	*Automatically pick the selected scenario did not work on already existing button widget*
UI *Calendar*	*DOC-895*	*[Calendar] The widget always displayed Instants in UTC even when changing time zone*
UI *Charts*	*DOC-1091*	*[TiP] [Charts] With legend placement set to ""bottom"*
	DOC-805	*[Chart] Tooltip was not ignoring formatting with time axis tooltip wasn't showing series names in split line chart*
	DOC-1043	*DateTime formatting in split by charts*
	DOC-1055	*The task codeReplicasUpdate was overriding customization of the Chart.yaml file*
	DOC-1088	*[Chart][Sort] Chart was not resolving BK categories correctly*
	DBPF-7849	*Tooltip did not support split by business key*
UI *Code Editor*	*DBPF-7590*	*CodeEditor configurator preview was broken*
UI *Data Grid / Explorer*	*DBPF-7294*	*[UI - Data Grid] Fill handle was causing an NPE in some cases when filling Dates*
	DBPF-7473	*Computed field columns were no longer grayed out in edit mode of Data Grid*
	DBPF-7474	*[Data Explorer] Column filters on dates and date time fields were not being displayed*
	DOC-924	*[Data Explorer] Filter on date-time fields from columns retrieved from another table did not display times*
	DOC-1019	*Pin columns in datagrid was not persistent*

UI *Gantt*	*DBPF-7026*	*[Gantt Configurator] Field selector automatic label was not always updated*
	DBPF-7067	*[Gantt] Light grey preset color wasn't taken into account*
	DBPF-7450	*[Gantt] Gantt context menu didn't close in some cases*
	DBPF-7458	*[Gantt] Gantt chart was not rendering properly with a numeric color-by field*
	DBPF-7492	*[Gantt] Grouping by a <NULL> value was not supported*
	DBPF-7512	*[Gantt / Chart] Custom color palette was not applied when using a scalar color-by field*
	DBPF-7570	*[Gantt] Resource selection could fail with ""overlapping events"" when zoomed in*
	DBPF-7572	*[Gantt] Zoom-out button was scrolling back to the left*
	DOC-1046	*[Gantt] Collapse of resource groups didn't work when customizing loadData*
	DBPF-7628	*[Gantt] Gantt resource selection was confusing resources with the same name*
	DBPF-7684	*[Gantt Configurator] Field selector automatic label was wrong when using business keys*
	DBPF-7738	*[Gantt] Editing scenario data was causing the Gantt to reload even if in edition*
	DBPF-7767	*[Gantt] Gantt's tooltip wasn't up to date for locally updated events*
	DOC-1103	*[Gantt] Gantt chart was losing scroll position after consecutive data edits*
	DBPF-7813	*[Gantt] Gantt was not parsing LocalDate field correctly*
	DBPF-7833	*[Gantt] Data edition was not supported when using LocalDate fields as start/end*

UI *Jobs*	*DBPF-7533*	*Clear button in job button configuration view was no longer visible*
	DOC-984	*Job List was ""jumping"" to the first page once any job status is updated*
	DOC-730	*[UI - Jobs] [Job Detail Widget] Line breaks were not respected for TEXT result output*
UI *Look & Feel*	*DBPF-6955*	*[Color picker] In some color pickers the color icons were editable text fields*
	DOC-1034	*The custom text color for the header was not applied on the widget toolbar*
	DBPF-7774	*[UI - Look & Feel] styling was inconsistent between text color and background color when using custom styles*
	DBPF-7872	*Modifying a widget configuration was not resetting the widget state in edition mode*
UI *Map*	*DBPF-7723*	*Map widget was ignoring configured colors*
UI *Scenario / Workspace List*	*DOC-994*	*It was impossible to empty the trash containing a locked scenario*
	DOC-975	*[UI - Scenario List] customizeGridOption was not taken into account in ScenarioListController*