SP19 - Dataiku Migration

The aim of this documentation is to provide key information on the migration of SP19 Dataiku projects such as :

- **Connections:** centralize connection information and contact points for each project to update future connection keys and certificates.
- **Production projects**: specify changes made to prod projects during migration.
- **Migration strategies** for each automation projects.
- **Non-Regression Test (NRT)** scenarios with **Solvay & Syensqo connections**.

This document describes the Dataiku migration projects “AS IS” **following** the Technical Design and Projects Assessment phases.

Pre-migration: Projects import from Solvay to Syensqo (1)

The goal is to transfer all Dataiku projects from Solvay to Syensqo's DSS instances (Design). For that, massive import script is used to transfer Self-Service and automation projects from Solvay DSS to Syensqo's DSS.

Project type: there are two types of project: **Automation projects** and **Self-service projects**

  - **Automation project (#7):** is a Dataiku project developed in Design environment, valited in UAT environment and deployed in Automation environment. Automation project is managed by a PO, Syensqo developer and DataOps team for deployment and production monitoring. In the SP19migration scope, OB team ensure all migration steps of **Automation projects** from Solvay to Syenqo's DSS instances (**Design UAT Automation**).

  - **Migration strategies**: are used to manage project version differences between Solvay's design and automation instances. When the difference between these instances is significant, the migration team, with the agreement of the project contact point, decides to migrate the Solvay automation version to the Syensqo design instance, and then migrate this production version to the Syensqo automation version via Syensqo's UAT instance. The Design version is also transferred to Syensqo's Design but is not routed to Syensqo's Automatin instance. If the project team wishes to put this version into production, it must ensure that it functions correctly in Syensqo's instances (Design, UAT and automation).

    Consulte this link to have more details about **Dataiku Migration Strategies**

  - **Self-service projects (#151):** is a Dataiku project existing only in design environment and managed only by a Syensqo Self-Service owner.

- **Import script:** Here the **link to access the import scripts** used for Dataiku projects migration.

These links to access SP19 Dataiku migration global information (Scope, data sources, scenarios, planning,...)

For more details about SP19 Dataiku projects migration planning check in the following link

The main **steps** of Dataiku projects migration from **Solvay to Syensqo physical environment** are the following:

Network Resquets: Solvay & Syensqo (2)

Opening ports, data flow, database access, obtaining access and certificate,... to etablish Syensqo contctions in Syensqo's DSS (Design/ UAT and Automation).

The global list of connection from Solvay's DSS of each instance is extracted to request openning flow and database acces to set up these connections in Syensqo DSS intances. Here the links to access thes connections list (Automation &nd Self-service projects):

- Automation projects connections lists:
  - From Solvay Automation instance:
  - From Solvay Design instance:
- Self-Service projects connections lists:
  - From Solvay Design instance:

Create Syensqo connection in design/UAT/prod (3)

New connections intended for use by Syensqo were created and tested across the three environments: test (Design), pre-production (UAT), and production (Automation). The login information from these new connections was used to update the configuration of the migrated connections currently used by the project.

This approach was preferred to allow modification of the connection objects only, without impacting the datasets. It also ensured that the original connection names were preserved, in line with the "AS IS" principle.

The newly created connections were used exclusively for the MySQL connections in the Mappy projects. This decision was made because Solvay operates only in pre-production and production environments, while Syensqo includes test, pre-production, and production. Using the same connection names across these differing environments would have led to confusion, as it would imply identical names for connections pointing to different servers.

Migration (4)

- **Migration Phase 1: Set up each project with Solvay connections + NRT in Syensqo's instance**

  In this step, Dataiku projects in **Syensqo** DEV environment will be **still connected** to their sources in **Solvay** environment. All connections of each automation project are test and fix from Dataiku graphical projects components, code envs (python code) and plugins.

  - CMIMPROVEMENT (Project's documentation)
    There are three scenarios to activate to make sure the project is running properly:

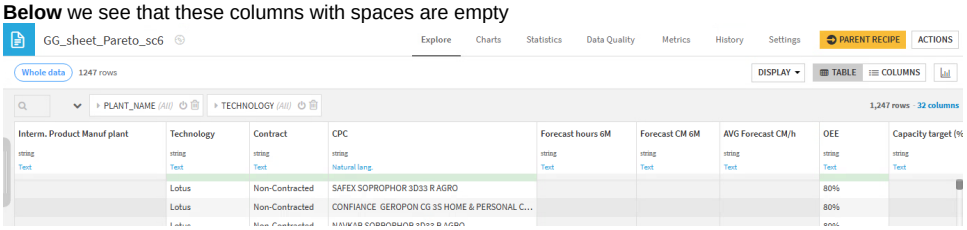
    After the connections are tested, and working, Monthy_reload_Data_Collection scenario is the most important to check since it is the baseline data for the whole project.
    The **Monthly_reload_Data_Collection** scenario is launched every 11th of every month, and fetches the data from BW sources.

Example: It is the 7th of may, the scenario hasn't run so the data from BW in the project is the data until the 11th of april, so the project is still using april's values to function.

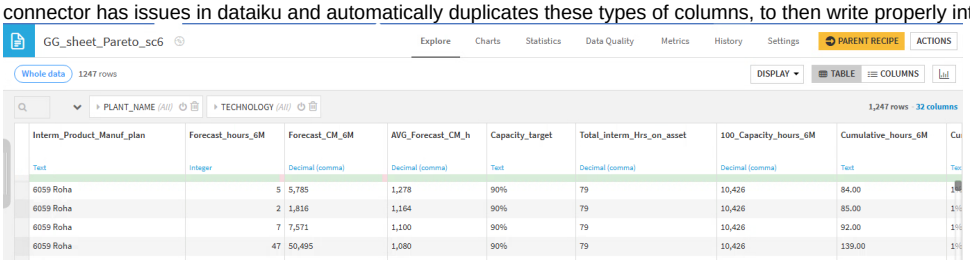
If it is the 12th of may, project will use the data from may because scenario will have ran.

WARNING: It is important not to manually relaunch the monthly scenario manually between the 1th of a month and the 11th of a month because of this, because the project will have data from a month that it is not expecting and it will cause a fail. This issue can be fixed by reworking some of the code in the following recipe:
When launching the scenario, the dataset tmp_SFA_BW_ds is partitioned by Month, which means it keeps an history of all previous months.

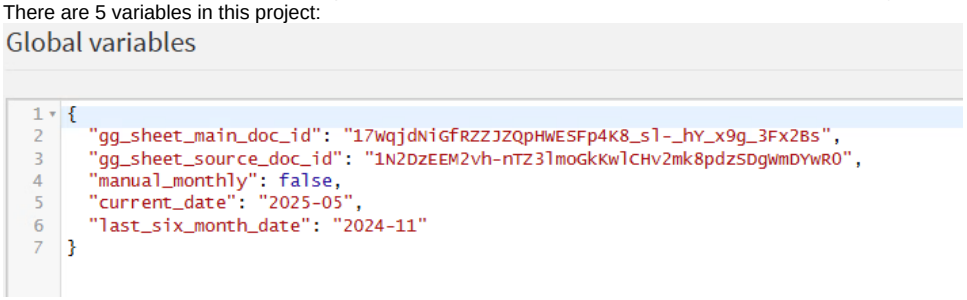
When installing the project on syensqo for the first time, to keep the history, the data was downloaded from solvay, then uploaded into syensqo following these steps:
The googlesheet schemas should not be touched, some of the column names have spaces instead of underscores, this can cause issues when trying to update schemas when the column names are too long.
Example: this dataset has 19 columns in reality, but it shows 32 columns, that is because columns with long names and spaces are duplicates into columns with underscores.
Below we see that these columns with spaces are empty



We can see **here** that we have the same column names but with underscores and which are filled, that is because the googlesheet connector has issues in dataiku and automatically duplicates these types of columns, to then write properly into googlesheets.

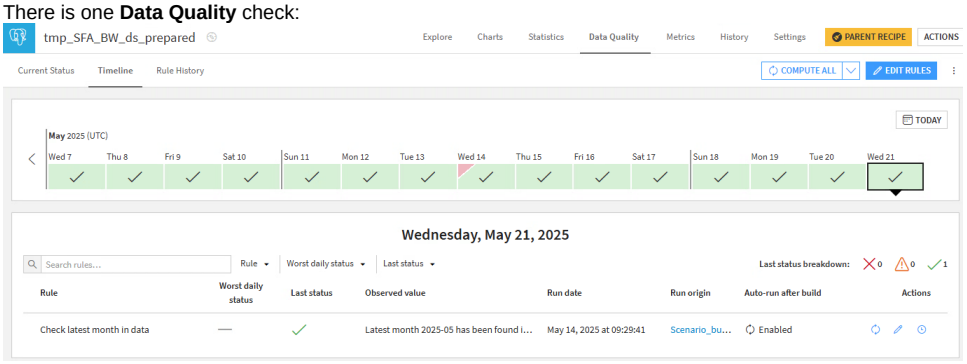


If the schemas is modified and the duplicated columns are deleted, then there will be no data in the gsheet for these columns.
There are 5 variables in this project:

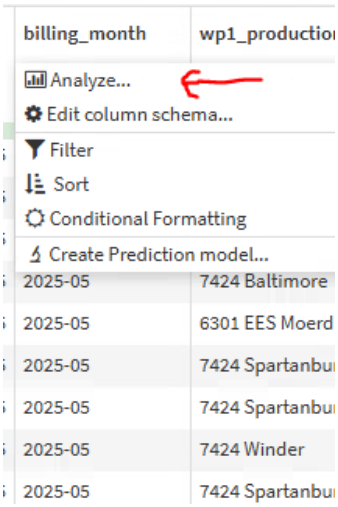


gg_sheet_main_doc_id is the id of the main googlesheet
same for second gsheet
manual_monthly is a variable that should not be touched
current_date is a variable that is updated automatically when the monthly_reload scenario is launched the 11th of current month, for example if it is the 9th of may, current_date will be 2025-04, if it is the 12th of may, it will be 2025-05
last_six_month_date is automatically updated too and is the history limit

There is one **Data Quality** check:



This DQ check verifies that max(billing_month) is equal to the variable current_date, in the dataset "tmp_SFA_BW_ds_prepared"
There is a dataiku issue that causes this check to fail after the monthly reload scenario even though the condition is met. It is fixed by going into the dataset after the monthly reload, and "analyse" the column "billing_month" which will then update the dataset correctly and fix the rule.



This issue might be a bug that has been reported to dataiku but for now it should be fixed this way.

To make sure that the project is working the same way as in Solvay instance, the output google_sheet output datasets are compared to the ones extracted in solvay, by checking the **volume of data** and the **data drift** for every column.
Link to code: https://dss.solvay.com/projects/TEST_USERS/notebooks/jupyter/drift%20comparator/

- NOVECAREDATACLEANING_V3 (Mappy) [Project's documentation]

The project was migrated from the Solvay automation node due to a significant delta between the design and automation environments. the design version was also transferred to Syensqo, using the _design suffix to distinguish it. During the migration, the connections were remapped to those intended for design use. Following the import, the Salesforce and Google Sheets connections were manually adjusted to align with the new configuration.

To validate the success of the migration, we relied on the activated scenarios in the production environment. Some zones and branches are no longer maintained and do not run even in the Solvay automation node; these elements were excluded from the testing process. Our validation focused on the following four key scenarios: run_all, send_to_gb, update_sfdc, and update_users.

Three out of the four scenarios executed successfully, with the exception of send_gbr. This scenario involves a Python recipe that generates a dataset using the pyrfc package, which is provided by SAP and is no longer actively maintained. Reinstalling this package required a custom setup beyond a standard pip installation, and identifying the correct configuration took considerable time.

After resolving the pyrfc issue, we encountered a network-related problem. Given that the generated dataset is not used by any other component in the project, a decision was made to exclude it from further consideration.

Warning:

Executing the full flow or certain zones within the flow may result in a crash due to the presence of recipes that use the same dataset as both input and output, which creates an infinite loop.

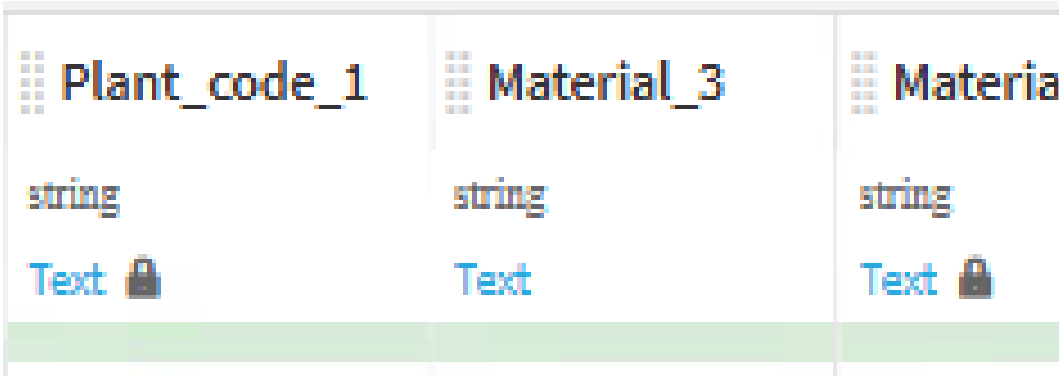
- GOOD_RECEIPT_SPLIT_PO2 (Project's documentation)

The project was migrated from the Solvay automation node due to a significant delta between the design and automation environments. the design version was also transferred to Syensqo, using the _design suffix to distinguish it. During the migration, the connections were remapped to those intended for design use. Following the import, the sap call in the scripts and the Google Sheets connections were manually adjusted to align with the new configuration.

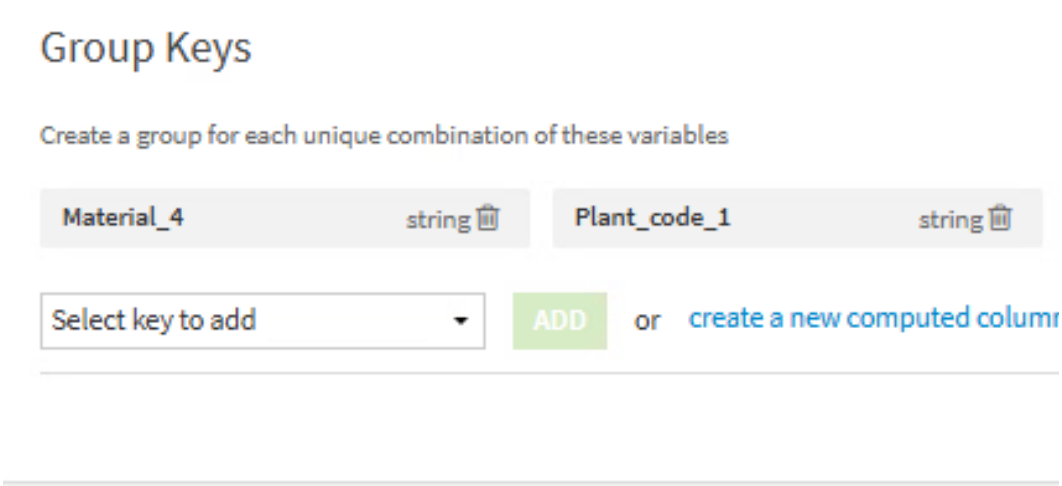
The scenarios run_monthly_history_D-5, run_monthly_history_D+2, Weekend's report, and Yesterday's report were used to validate the migration. It is important to note that the project contains two distinct flow branches: one for Solvay data and another for Syensqo data. For the purpose of this validation, only the Syensqo branch was considered, and the Solvay branch was excluded. All selected scenarios executed successfully, confirming the integrity of the migration.

Warning:

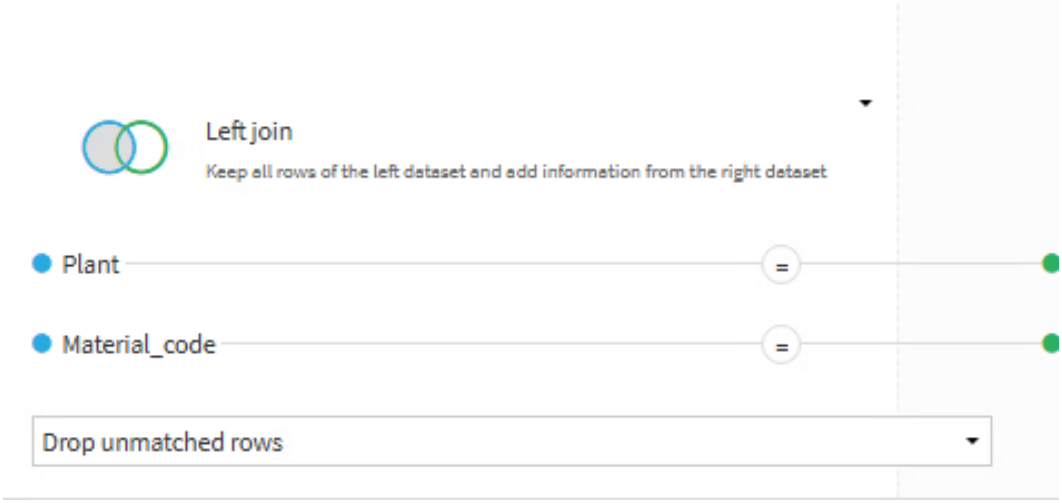
- if during a run and for any reason the sap retreived datasets are empty. the following execution will crash if the recompute schema checkbox is checked. to make it run again you need to:
  - force the types of the following columns in the compute_Material_rates_p



- Check if the compute_MMD_price is configured as follow



- check if the compute_PO_price is as follow



  - The compute_Material_Master_rates_p prepare recipe applies a filter on the Valuation_class column. this filted values changed from solvay to syensqo.

- PCM_V2_SPLIT_PO2 (PCM) [For futher information about this project migration please contact project team]
- SMARTCHEM
  This project consists of multiple projects, Molecule_search_v2_sco, smartchem_v3_sco, and chemspace_4
  These projects do not have any connection to configure, they have data that has been manually uploaded to the instance and serve as a baseline for webapps.

All that was needed was to verify that the data uploaded between the solvay instance and the syensqo instance are the same, and that the webapps are working properly.

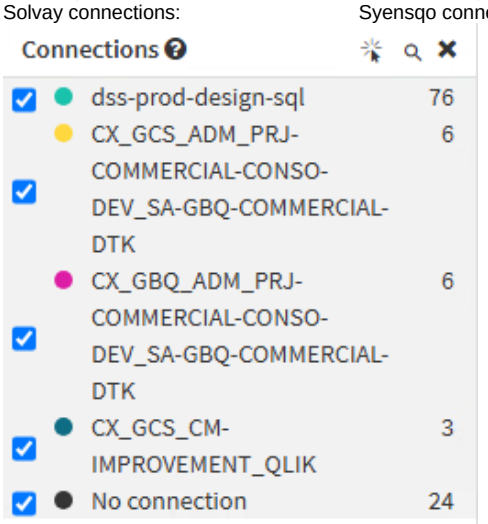
- SPOT_SPP_MASTER [(Project's documentation) For any information about this projet migration contact the project team]
- SPP_SALES_FORECAST_SCO2 (DISCARDED)

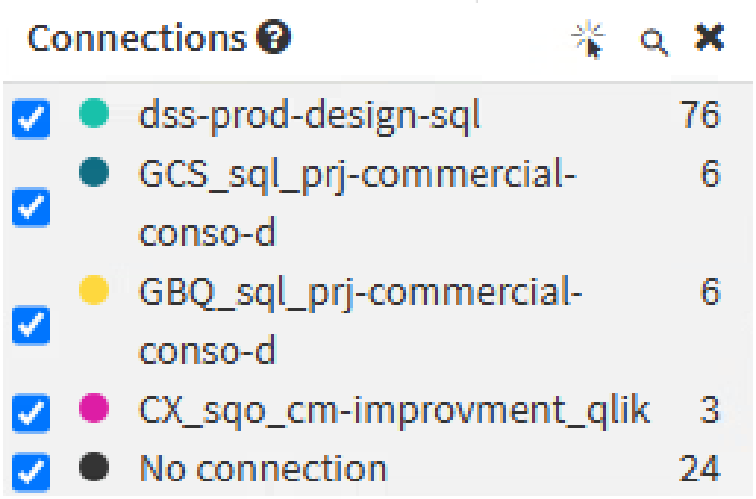
- **Migration Phase 2: Set up each project with Syensqo connections + NRT in Syensqo's instance**

In this step, all connections in Syensqo's DSS design are swicht on Syensqo's one. For that the equivalent of Solvay connections are set up with Syensqo's ressources created with flow openning.

- CMIMPROVEMENT

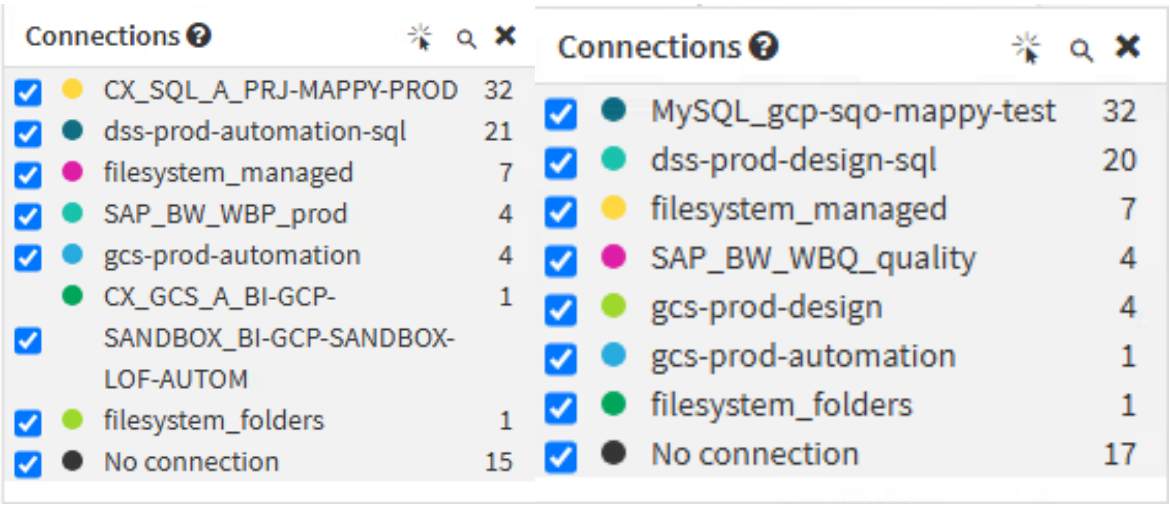
For this part, connections are switched to their syensqo equivalent.
Solvay connections: Syensqo connetions:





The "No connection" dataiku recipe refer to googlesheet recipes, these recipes need to be manually modified so they can point towards the right sheet.
The syensqo sheets created for the dev/uat are inside "Tool CMI Novecare" in the
(Project's documentation)
and all have the prefix **Syensqo UAT/DEV CMIMPROVEMENT**
Similarly to what was done with the project's run on solvays connection, the output data with syensqo connection is compared to solvays output data to make sure there are no changes.

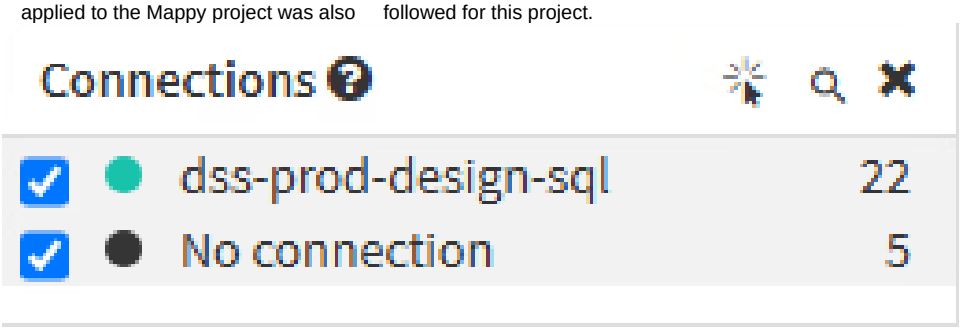
- NOVECAREDATACLEANING_V3 (Mappy)The project was migrated from the automation node, and the connections were mapped to their corresponding design connections. The dataset output_mail_pf1_corporate_group, which relied on the dss-prod-design-sql connection, was deleted since its parent recipe had already been removed in the production environment.Solvay connections : Syensqo connections:



Following the migration, all connections were updated to point to the new Syensqo data sources.
Solvay's dss-prod-design-sql:

Syensqo's dss-prod-design-sql:

- GOOD_RECEIPT_SPLIT_PO2
  In addition to a few Google Sheets, this project exclusively uses the `dss-prod-design-sql` connection. The same approach that was applied to the Mappy project was also followed for this project.



- PCM_V2_SPLIT_PO2 (PCM) [For any information regarddind this projects migration please contact project team]
- SMARTCHEM N/A (Only migrated on Syensqo's Design instance)
- SPOT_SPP_MASTER [For any information regarding this project migration please contact: project team]
- SPP_SALES_FORECAST_SCO2 (DISCARDED)

Test phase (5)

- NRT Test: is a technical test carried out by devOps based solely on the comparison of data between Solvay and Syensqo DSS instances..
- UAT Tests (business)
  - CMIMPROVEMENT: For the test, the project is pushed to the UAT environment, and all the scenario are launched to verify that the whole project is working correctly. The project owner then checks if everything is behaving accordingly (Data is ok, webapp is working) and then it is possible to go to Automation Environment.

- GOOD_RECEIPT_SPLI_PO2:
  For testing purposes, the project is deployed to the UAT environment, where all scenarios are executed to ensure the project functions as expected. Once validated by the business team and confirmed to be behaving correctly, the project can then be promoted to the Automation environment.

Deployment (Automation Env.) & technical test (6)

Deployment is only carried out for automation projects after transfer from Solvay's Design instance to the Syensqo design instance, migration and testing phases in the Syensqo UAT instance. After business validation, projects are moved from the Syensqo UAT instance to the Syensqo automation instance. All connections are updated to use Syensqo automation resources (data sources: BW, BigQuery, Salesforce, ProsgreSQL, etc.).

- **Sanity check before BW Go Live:**

  After deploying projects in automation, the aim of this stage is to ensure that all connections to Syensqo's data sources are operational, except for the BW source.

- **Sanity check after BW Go Live:**

  Once the BW source has been commissioned, the aim is to ensure that all connections to Syensqo's BW data sources are operational, and then to test the project scenarios or planned schedule to test the end-to-end execution of the ech automation project.

Self-service projects migration & support (7)

As part of the SP19 Dataiku migration, Self-Service projects are migrated from Solvay's Design instance to Syensqo's Design instance. Only the main connections to Syensqo resources are set up by the migration teams, while the remaining tasks must be carried out by the project owners. The migration teams provide back-up for all self-service project owners in setting up their projects and resolving connection problems.

Here the list of dataiku Self_Service projects migrated in SP19 Dataiku projects migration.

Go Live (Syensqo) (8)