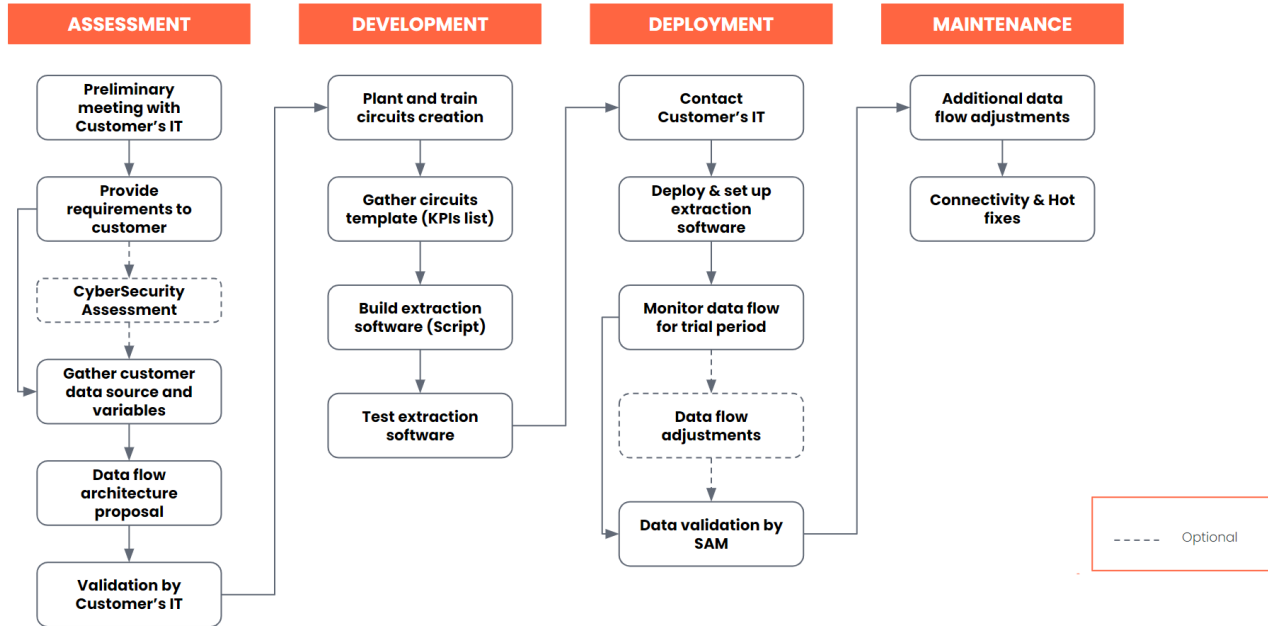


Customer Onboarding

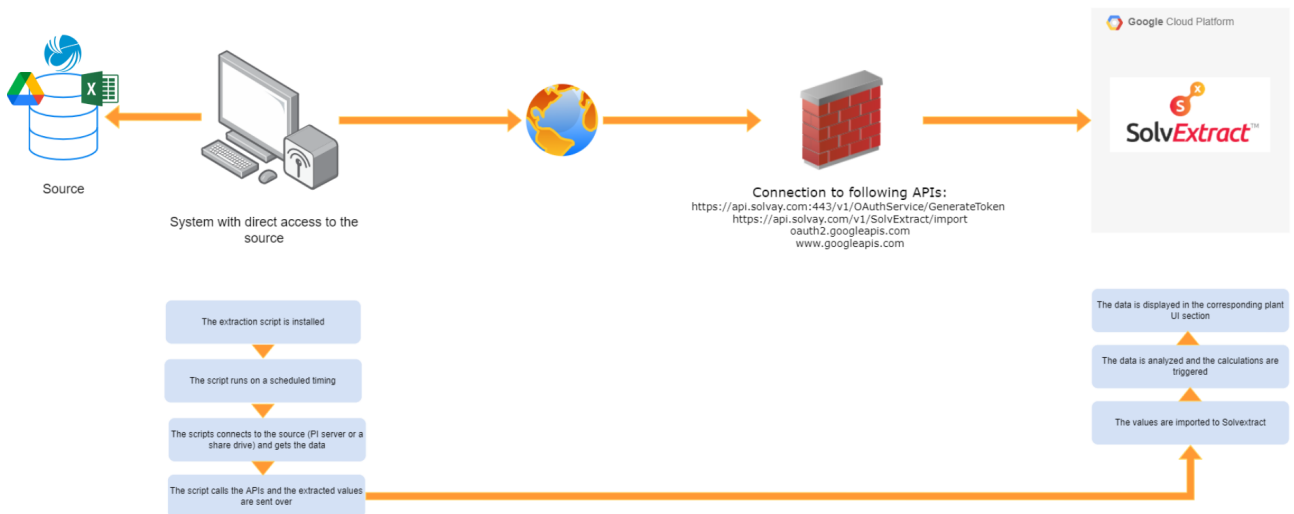
Onboarding process

Onboarding Process



Data Connectivity

High Level Overview



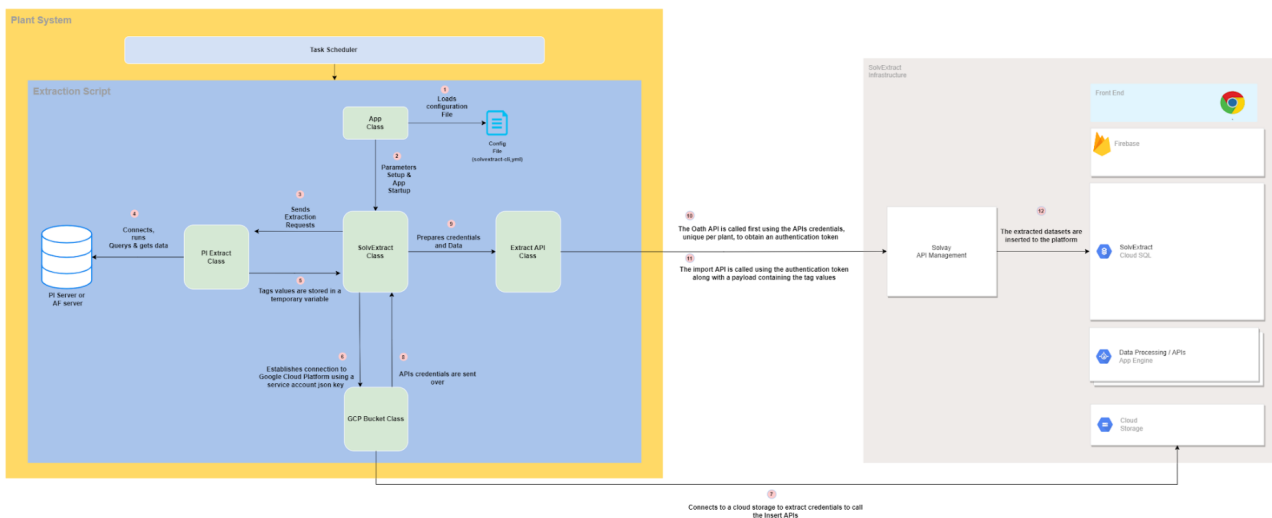
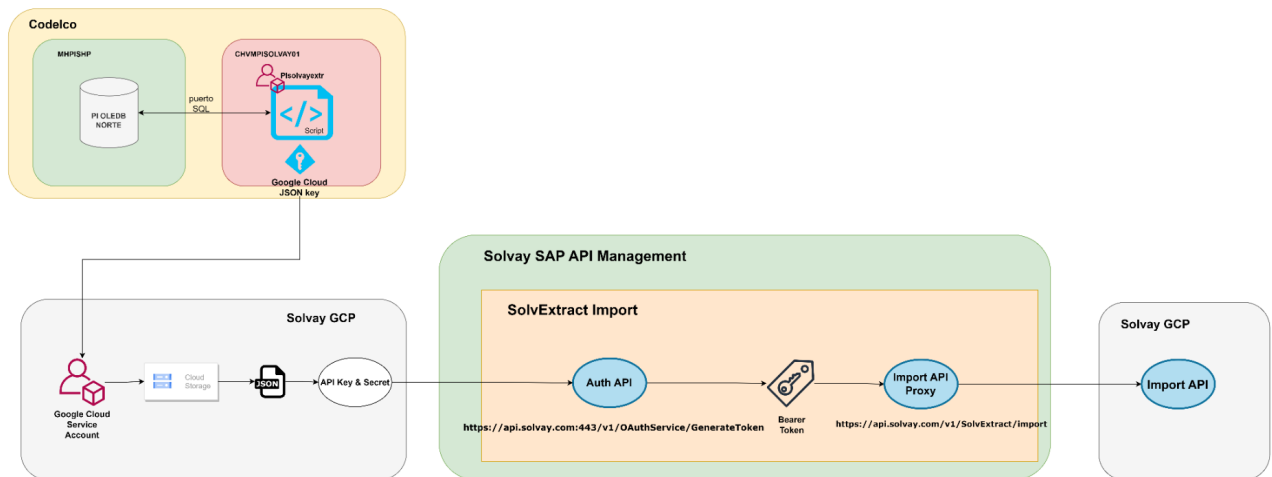
A script or program is provided and installed in the customer's designated system or machine. This script will connect locally to the customer's select data source, extract the data values and send them over to the SolvExtract platform via an API. Nonetheless, there are other instance when the extraction method is hosted and managed solely on the Syensqo's side.

Extraction Methods

The customer must provide information on how the data will be shared. Currently, the digital mining team supports the following options as data sources:

- PI server
- Excel spreadsheet(s) located in one or multiple network drives
- File uploaded to a customer owned Shared Google Drive
- File shared as an attachment via e-mail
- File uploaded to a Syensqo owned GCP bucket

PI Method



Requirements

If the selected data source is either **PI server** or **Excel spreadsheets**, then a script will be provided and thus the customer must open firewalls and ports to the following:

- oauth2.googleapis.com
- googleapis.com
- <https://api.solvay.com:443/v1/OAuthService/GenerateToken>
- <https://api.solvay.com/v1/SolvExtract/import>
- **Port: 443**

The user must install the following libraries in their system where the script is deployed:

- Visual C++ 2015 Redistributable
- PI OLEDB Enterprise

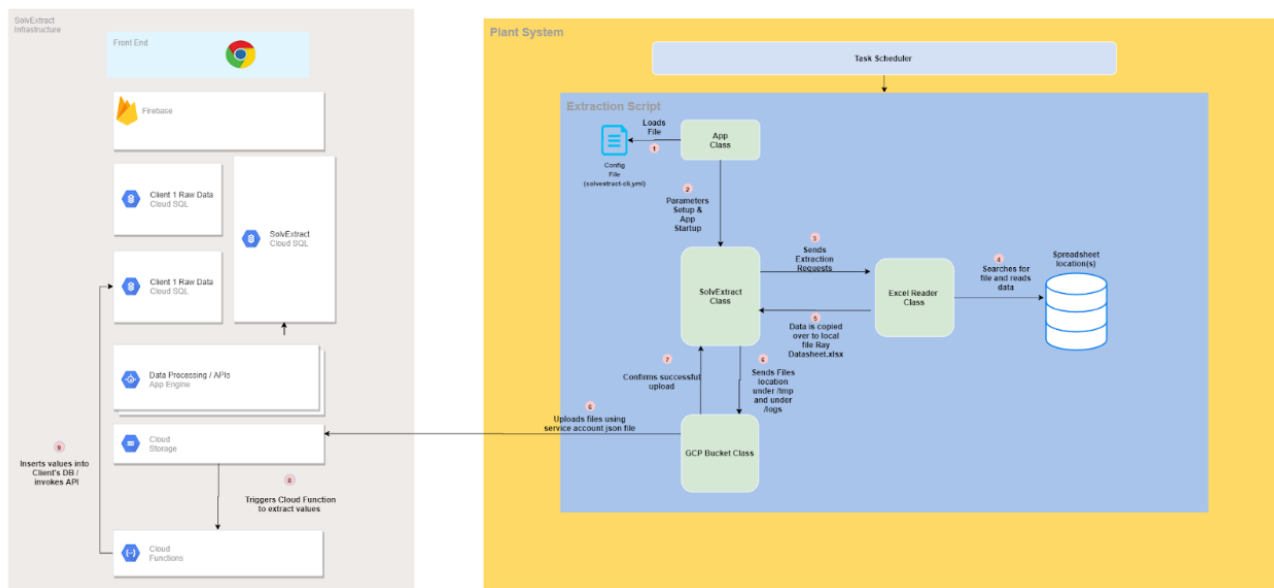
Setup

- A script will be provided and should be deployed in a machine capable of accessing the data using either of the below proposed methods.
- The script must be put into a cron job using task scheduler(windows) thus it runs at least once per day.

Process

- When the script is kicked off, the App file will load the solvextract.yml. This file contains all the configuration needed to run the script such as PI server name, the database, credentials (if needed), the time zone, the variables and tags that need to be extracted.
- The App file will then call the Solvextract file, which will be in charge of assigning the configuration values to variables, calling the PI Class to connect to the PI server and run the query, creating the connection to Google Cloud Platform, calling the Extract API class to send over the data.
- The PI Extract file is in charge of connecting to the PI system locally (via localhost or using a server name or IP), running a query using the kpis tags, getting the values back and storing them in a variable.
- When the PI Extract completes its task, the Solvextract File will then create the connection to the Google Cloud Platform using a service account json key file using the GCP Bucket Class.
- The GCP bucket will connect to a Cloud Storage directory where the Insert APIs key and secret are stored. These credentials are unique per customer/client.
- Then the Insert API class will first call the OAth API using the credentials in order to obtain a secure token that expires after 3 minutes.
- Once the token is successfully obtained, the Insert API class will call another API using the token along with a json payload that contains the extracted tag values.
- After the API is invoked, the Solvay API management will redirect the API to another internal API which will validate the token and then insert the data accordingly into the Solvextract Platform.
- And finally, the script log file is uploaded to the cloud storage and PI, API and Google Cloud Platform connections are closed afterwards.

Excel Extraction Method



Requirements

If the selected data source is either **PI server** or **Excel spreadsheets**, then a script will be provided and thus the customer must open firewalls and ports to the following:

- oauth2.googleapis.com
- googleapis.com
- <https://api.solvay.com:443/v1/OAuthService/GenerateToken>
- <https://api.solvay.com/v1/SolvExtract/import>
- **Port: 443**

The customer must also provided:

- Name of the file(s)
- Full network path(s) where the file(s) are located.
- File(s) location (If file(s) is located in a network drive then the task scheduler running the script must use an account that's able to access and read the file(s))
- Samples of the file(s) in order to see the format and adapt the script so it can read and extract the needed values.

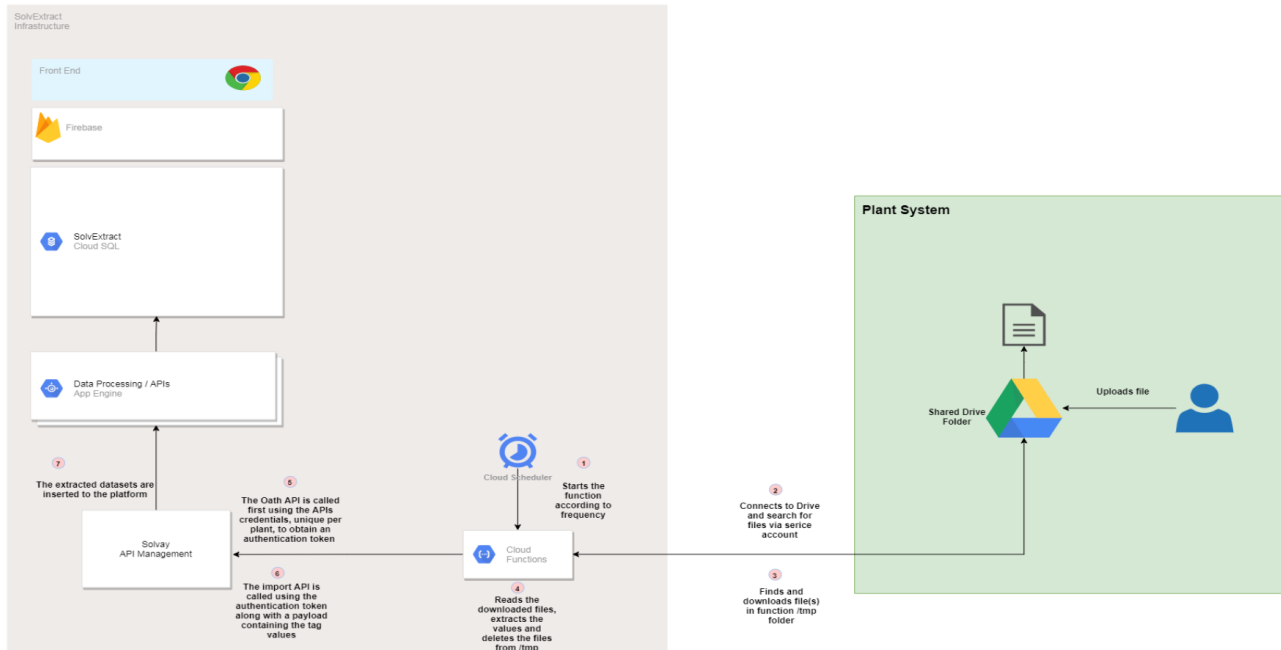
Setup

- A script will be provided and should be deployed in a machine capable of accessing the data using either of the below proposed methods.
- The script must be put into a cron job using task scheduler(windows) thus it runs at least once per day.

Process

- When the script is kicked off, the App file will load the solvextract.yml. This file contains all the configuration needed to run the script such as the location of the network drive, the file name to be read, the timezone and the variables that need to be extracted.
- The App file will then call the Solvextract file, which will be in charge of assigning the configuration values to variables, calling the Excel reader, establishing the database connection and insert the data and finally calling the GCP bucket file to start the connection to our Cloud platform to upload the logs file.
- The Excel Reader file is in charge of searching and finding the customer's file(s), that could be located locally or in a network folder, needed to get the kpis values. It will also validate whether or not the directory or the file is available throwing an error if not found.
- If the file is found, the excel reader will make a local copy of this file under /temp/, read it, extract the values accordingly, put those values in a variable and finally delete the copied file.
- When the PI Extract completes its task, the Solvextract File will call the APIs. The first API validates the unique credentials and api key in order to generate a token. This token will be used in the next API call and it will send over the extracted data and insert it into the Cloud database.
- And finally, the GCP Bucket function will upload the logs to our cloud storage.

Google Drive Method



Data Collection

Navigation tree

[Expand all](#) [Collapse all](#)

Pages recently viewed