

DFS TD - Synthesis Drying Raw Data

Technical documentation for the data coming from drying tests

Summary

- [Sum-up](#)
- [Data Sources](#)
- [Data Collection](#)
 - [Schema using Google Drive](#)
 - [Schema with file share](#)
- [Data Preparation](#)
 - [Parse](#)
 - [Compute](#)
- [Presentation](#)

Sum-up

Equipment / Scale	Tesla	Gunsan	Colognes 2500L
Data Sources	ELN , Raw Data on Google drive	ELN , Raw Data on file share	ELN , Raw Data on Google Drive
Raw Data File type	CSV	xlsx	xls
Scale Name on ELN	FR-170L-TESLA	KR-170L	FR-2500L
Data Collection	Talend: R011_Download_Synthesis_gDrive_Drying	Talend: J010_Download_Synthesis_LabServers Python : download_drying_gunsan.py	Talend: J011_Download_Synthesis_gDrive_Drying
Parse	Python: parse_drying_tesla.py	Python: parse_drying_gunsan.py	Python: parse_drying_2500L.py
Compute	Python: compute_drying_tesla.py	Python: compute_drying_gunsan.py	Python: compute_drying_2500L.py
BigQuery	Target tables: • <code>raw_data_synthesis.DryingDetails</code> • <code>raw_data_synthesis.DryingSummary</code>		
Mapping spreadsheet	Drying Mapping		

Data Sources

- [ELN](#)
- Raw Data on Google Drive

Data Collection

The talend jobs **J010_Download_Synthesis_LabServers** and **J011_Download_Synthesis_gDrive_Drying** extract the raw data files listed on the ELN table **drying_raw_data_link** for which the field "drying_equipment_name" is the scale name, i.e. "FR-170L-TESLA".

Schema using Google Drive

Examples

Talend jobs

- R011_Download_gDrive_Drying
- R012_Download_gDrive_Filtration

Tmp Folder

- D:\DATA\[ENV]\Rn\Silica\tmp\Synthesis\DryingTesla
- D:\DATA\[ENV]\Rn\Silica\tmp\Synthesis\Filtration2500L

Schema with file share

Examples

Lab servers source

- \\FRPH2-labpc-backup\labo\W-522649\DATAS DATALAKE
- \\FRPH2-LABPC-BACKUP\LABO\W-509931

Python files

- download_filtration170L.py
- download_synthesis25L.py

Output folders

- D:\DATA\[ENV]\Rn\Silica\tmp\Synthesis25L
- D:\DATA\[ENV]\Rn\Silica\tmp\Synthesis170L



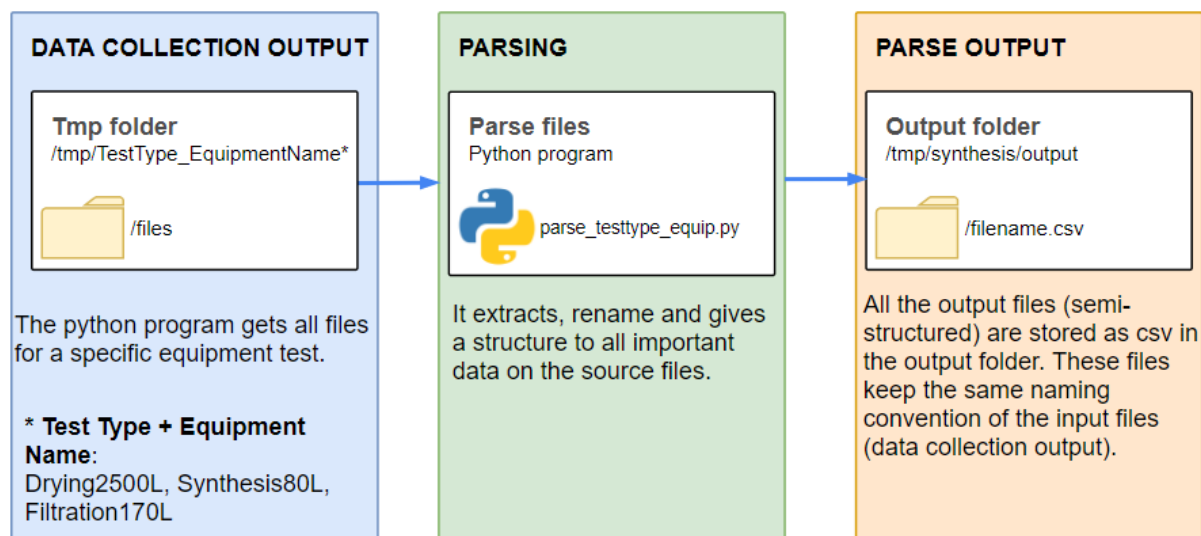
Please refers to the [DFS TD - Synthesis - Norms and Conventions](#) for the output filename convention on the Data Collection section

Data Preparation

Parse

The parsing python scripts extracts from the raw data files the needed columns.

DATA PREPARATION - PARSE



Columns List

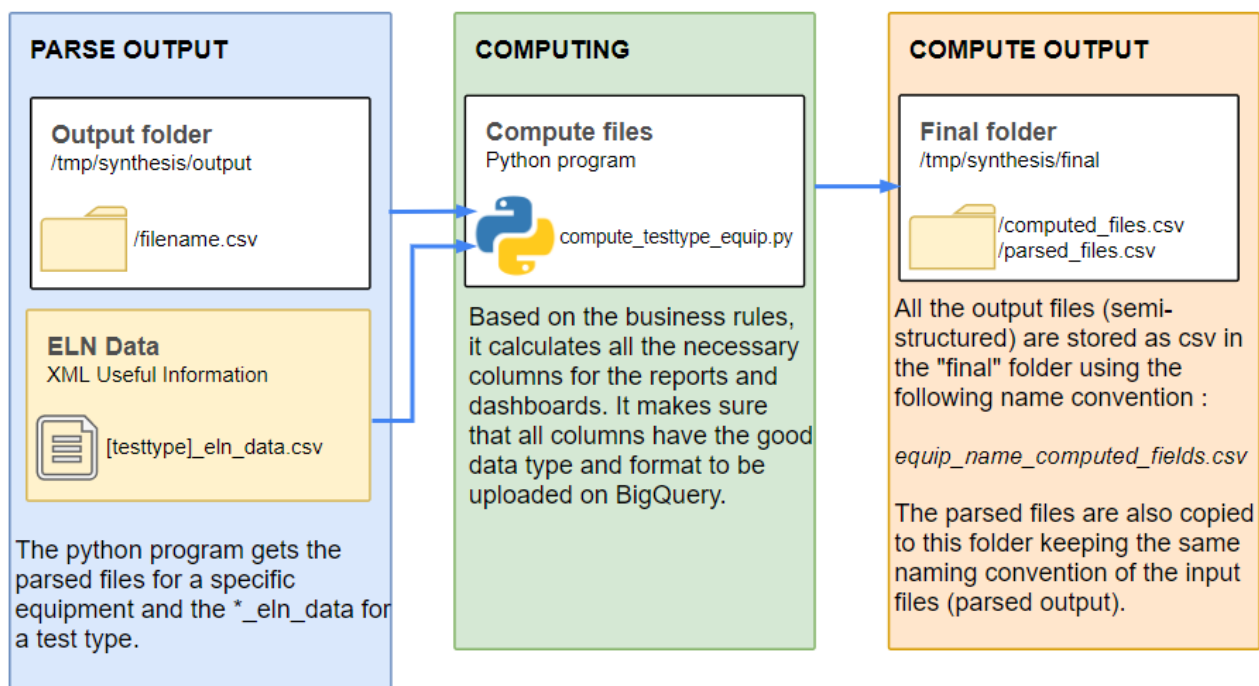
For each sample, the script extracts the many fields from the raw data files and outputs a .csv file. For the mapping details, please refers to the sheet "Parse Mapping " on the Drying Mapping spreadsheet (link to the spreadsheet on the Sum-up section).

Compute

The compute python script uses as input the parsed .csv files previously created . It computes the new columns and values from raw data and regenerates new files.

If the output files already exist the script will **NOT** replace them.

DATA PREPARATION - COMPUTE



In the beginning of the script , it extracts many columns or values from the file **drying_eln_data** . Those values are used in later computations as constants.

For each sample, it creates two different files that will be used to create new tables on BigQuery :

DryingDetails

The **first table** is composed of the columns previously extracted from the raw data files and the new columns calculated during the execution.

Dataset : raw_data_synthesis_mig

For the columns details, please refers to the sheets " **Details Mappings** " on the **Drying Mapping** spreadsheet (link to the spreadsheet on the Sum-up section).

DryingSummary

The **second table** is composed of the new values computed from raw data. This is a atomic table and it aggregates the values by **unique_id, study_id and sample_id** which represents one line per data raw file.

Dataset : raw_data_synthesis_mig

For the columns details, please refers to the sheets " **Summary Mapping** " on the **Drying Mapping** spreadsheet (link to the spreadsheet on the Sum-up section).

Presentation

The details and summary files are created as tables on **BigQuery** unifying all scales in the same tables. A Talend job is responsible to push all this data to a dataset called **raw_data_synthesis_mig**.

