## **Overview**

This code is designed to perform data processing and analysis on wellbore data using Apache Spark. It focuses on data cleaning and visualization of time series data related to various parameters of wellbore operations.

## **Prerequisites**

* Python 3.9
* Apache Spark
* PySpark
* Pandas
* Matplotlib

## **Usage**

1. Download the file ***geolog\_final.ipynb***
2. Place the sample files, that is these 5 files:
   1. “Norway-Statoil-NO 15\_$47$\_9-F-1 C 0 time.csv",
   2. “Norway-Statoil-NO 15\_$47$\_9-F-11 time.csv",
   3. "Norway-Statoil-NO 15\_$47$\_9-F-5 time.csv",
   4. "Norway-Statoil-NO 15\_$47$\_9-F-7 time.csv",
   5. "Norway-Statoil-NO 15\_$47$\_9-F-4 time.csv"

in the data folder using the links provided in email.

1. Make sure you have all the prerequisites installed.
2. Open the file in jupyter notebook
3. Run the code.

## **Spark Integration**

Apache Spark is utilized in this code for efficient distributed data processing. It enables scalable and parallelized operations on large datasets, making it suitable for handling big data tasks.

## **Data Cleaning Process**

The data cleaning process involves several steps to ensure the data is properly formatted and relevant for analysis:

* Reading CSV files into Spark DataFrames.
* Renaming columns for consistency.
* Filtering out irrelevant columns.
* Extracting relevant information from columns using regular expressions.
* Adding additional columns for year, month, and day extracted from the date-time column.
* Selecting desired columns for further analysis.

## **Visualization**

The code includes functionality to visualize time series data for specific parameters using Matplotlib. Time series plots are generated for each parameter, providing insights into the trends and patterns over time.

## **Challenges Faced**

* Lack of Data Knowledge: The data provided posed a challenge as I had no prior knowledge about its structure or meaning. This made it difficult to identify important fields and key performance indicators (KPIs) for analysis. Future iterations of this project could benefit from a thorough explanation of the data and its context to improve analysis.
* Airflow Installation: I encountered difficulties installing Apache Airflow for scheduling tasks. Lack of prior experience with Airflow installation and configuration hindered progress in automating data processing tasks.
* AWS S3 Integration: Although the current implementation involves manual downloading of data files, the intention is to store these files in AWS S3 and access data directly from there. Due to time constraints, this integration was not implemented in the current version. Future enhancements will involve setting up AWS S3 integration for seamless data access and storage.