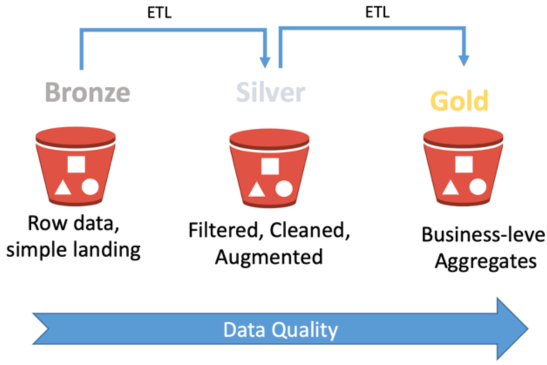
Movielens Data Analysis

Learning objectives:

The learning objectives of this lab are:

- Create an **ETL pipeline** with Apache Spark using the movielens dataset.
- Practice batch and ad hoc processing with Spark and Hive.
- Create and manage Hive tables and jobs with Hive CLI



Data set:

- This dataset (ml-25m) describes 5-star rating and free-text tagging activity from MovieLens, a movie recommendation service. It contains 25000095 ratings and 1093360 tag applications across 62423 movies. These data were created by 162541 users between January 09, 1995 and November 21, 2019. This dataset was generated on November 21, 2019.
- Users were selected at random for inclusion. All selected users had rated at least 20 movies. No demographic information is included. Each user is represented by an id, and no other information is provided.
- The data are contained in the files genome-scores.csv, genome-tags.csv, links.csv, movies.csv, ratings.csv and tags.csv. More details about the contents and use of all these files follows.
- More information on

https://grouplens.org/datasets/movielens/

ETL Pipeline with Apache Spark

- **We want to create an ETL pipeline with spark:**

EXTRACT

- Load the data into your Hadoop cluster and create the necessary dataframes
- Explore the different DFs

TRANSFORM

- Analyze the movielens data, do the necessary processing and transformations and create a new dataset, in which we have the following fields:
  - *Movie ID*
  - *Movie name*
  - *Year of release*
  - *Number of ratings*
  - *Genre (for every movie you might add several lines, 1 line per genre)*
  - *Rating average*

LOAD

- Load the new dataset into a parquet file (or CSV)
- **Provide necessary screenshots for your code and the output.**

Data Set Analysis with Spark

- Load the new silver dataset (CSV or parquet) into apache Spark and create a dataframe. Then using Spark DataFrames and SQL, write the following queries:
  - Best movie per year
  - Best movie per genre
  - Best 'action' movie per year
  - Best romance per year
- **Provide necessary screenshots for your code and the output.**

Data Set Analysis with Hive

Create 2 external Hive tables (for movies and ratings). And then answer the same questions using hive queries:

- Analyze the movielens data, do the necessary processing and transformations and create a new hive table, in which we have the following fields:
  - *Movie ID*
  - *Movie name*
  - *Year of release*
  - *Number of ratings*
  - *Genre (for every movie you might add several lines, 1 line per genre)*
  - *Rating average*

- Best movie per year
- Best movie per genre
- Best 'action' movie per year
- Best romance per year

- **Provide necessary screenshots for your code and the output.**

Batch Processing

Write the previous steps to create the ETL pipeline and data analysis in 2 scripts, one for the Spark job and another for the Hive job. Let’s call these scripts:

  - ○ spark-etl.py
  - ○ hive-etl.hql
- execute them on the cluster

Submission Deadline:

- 14 January 2026 midnight.
- Work in pairs
- One PDF file with necessary screenshots and explanations, in addition with the scripts.