Project 5

**Overview**

The goal of the project is to build a time series analysis pipeline of a Covid19 dataset.

The system must provide distributed computing capabilities using Apache Spark middleware.

Being the problem a batch processing on a given dataset, we decided to use SparkSQL to develop the processing.

The overall computation is handled by one big Spark driver program which provides result for each of the three subqueries required.

To provide protection against crashes, each subquery is persisted, when computed, by Spark workers.

**Query 1**

Makes a windowing function

1. partitioning by country,
2. ordering by day,
3. aggregating in the moving average period.

Finally, an average of cases over the window is computed

**Query 2**

Makes a windowing function

1. partitioning by country,
2. ordering by day.

A new daily lagged column of previous query is then added to the data, replacing null border values with not lagged values.

Result is obtained by aggregating (dividing) previous query and previous lagged attribute.

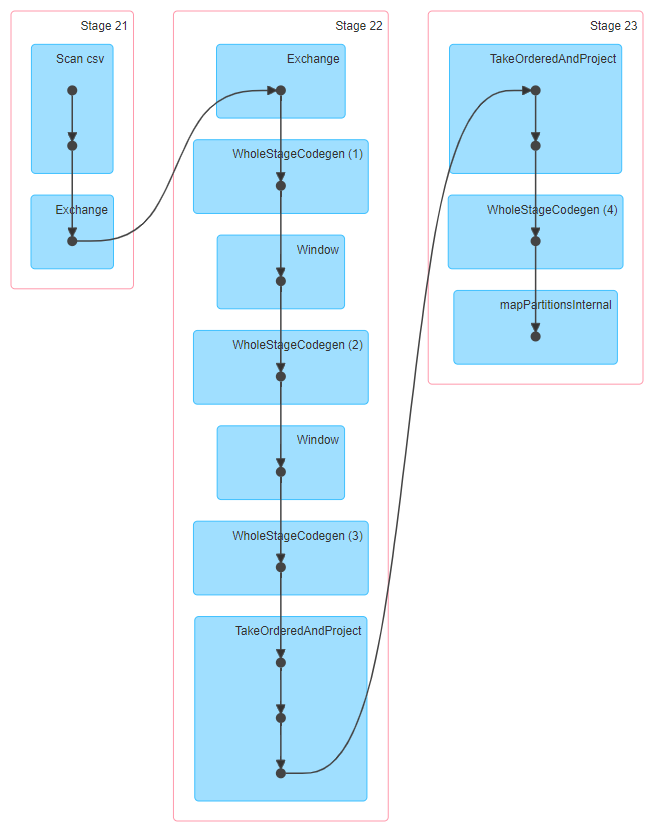
**Query 3**

Data is

1. grouped by country,
2. aggregated by sum of previous query attribute,
3. ordered (descending) on previous aggregated attribute.

Finally, top n countries are selected.

Stages of the overall computing pipeline



**Performance analysis**

When increasing only n\_days, execution time does not change.

When increasing n\_countries (and optionally n\_days), execution time is (at most, approx.) n\_workers times faster than a serial execution.

When increasing number of workers, if n\_countries >= n\_workers then execution time is (at most, approx.) n\_workers times faster than a serial execution.