Part 1: Apache Spark

Miha Torkar
Jozef Stefan Institute
17th October 2019
miha.torkar@ijs.si
Overview

• What is Apache Spark?
• Main components
  • Resilient Distributed Dataset (RDD)
  • DataFrames
• Initializing Spark on your computer
• First examples
• SQL type queries
What is Apache Spark?

• Unified computing engine and a set of libraries for parallel data processing on computer clusters

• Unified:
  • One platform for all applications & tasks

• Computing engine:
  • Handles loading data from storage systems
  • Performing computation on it
  • Not permanent storage as the end itself

• Libs: SQL, streaming, ML

• Supports multiple programming languages (Scala, Java, Python and R)
Resilient Distributed Dataset (RDD)

- Low level object
- Splitting data across multiple nodes in the cluster
- Create RDD: parallelize existing collection or reference existing storage system (HDFS, HBase, any other Hadoop Input Format)
- Caching dataset in memory
  - Different storage levels available
  - Fallback to disk possible
Spark RDD Operations

- Transformations build RDDs through deterministic operations on other RDDs:
  - Transformations include *map, filter, join, union, intersection, distinct*
  - Lazy evaluation: Nothing computed until an action requires it

- Actions to return value or export data
  - Actions include *count, collect, reduce, save*
  - Actions can be applied to RDDs
  - Actions force calculations and return values

- When to use RDDs?
  - You need some functionality not present at higher level APIs (e.g. tight control over physical data placements across the cluster)
  - Maintain legacy codebase written using RDDs
  - Custom shared variable manipulation
Spark DataFrames

- RDD hard to work with directly, so using the Spark DataFrame abstraction instead
- Spark DataFrame was designed to behave a lot like a SQL table
- can run SQL queries on the tables
- Conversion from other data types
- Storing locally or on cluster
Spark DataFrames

Hive Data
Csv Data
Json Data
RDBMS Data
XML Data
Parquet Data
Cassandra Data
RDDs

<table>
<thead>
<tr>
<th>Col1</th>
<th>Col2</th>
<th>Col3</th>
<th>.....</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Row 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Row 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.....</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
How does execution look like?
Initializing Spark

- Directly from terminal

- Inside Jupyter notebook:
  - import pyspark,
  - Start SparkSession
    - Main entry point for Spark functionality
    - Connection to a Spark cluster
    - Can be used to create RDDs and to broadcast variables to cluster
    - Online documentation

- Spark UI
Spark DataFrames
DataFrame Transformations

- Remove columns or rows
- Transform a row into a column or a column into a row
- Add rows or columns
- Sort data by values in rows