Using BigBench to compare Hive and Spark

Alejandro Montero, Nicolas Poggi

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

BSC Microsoft Research Centre

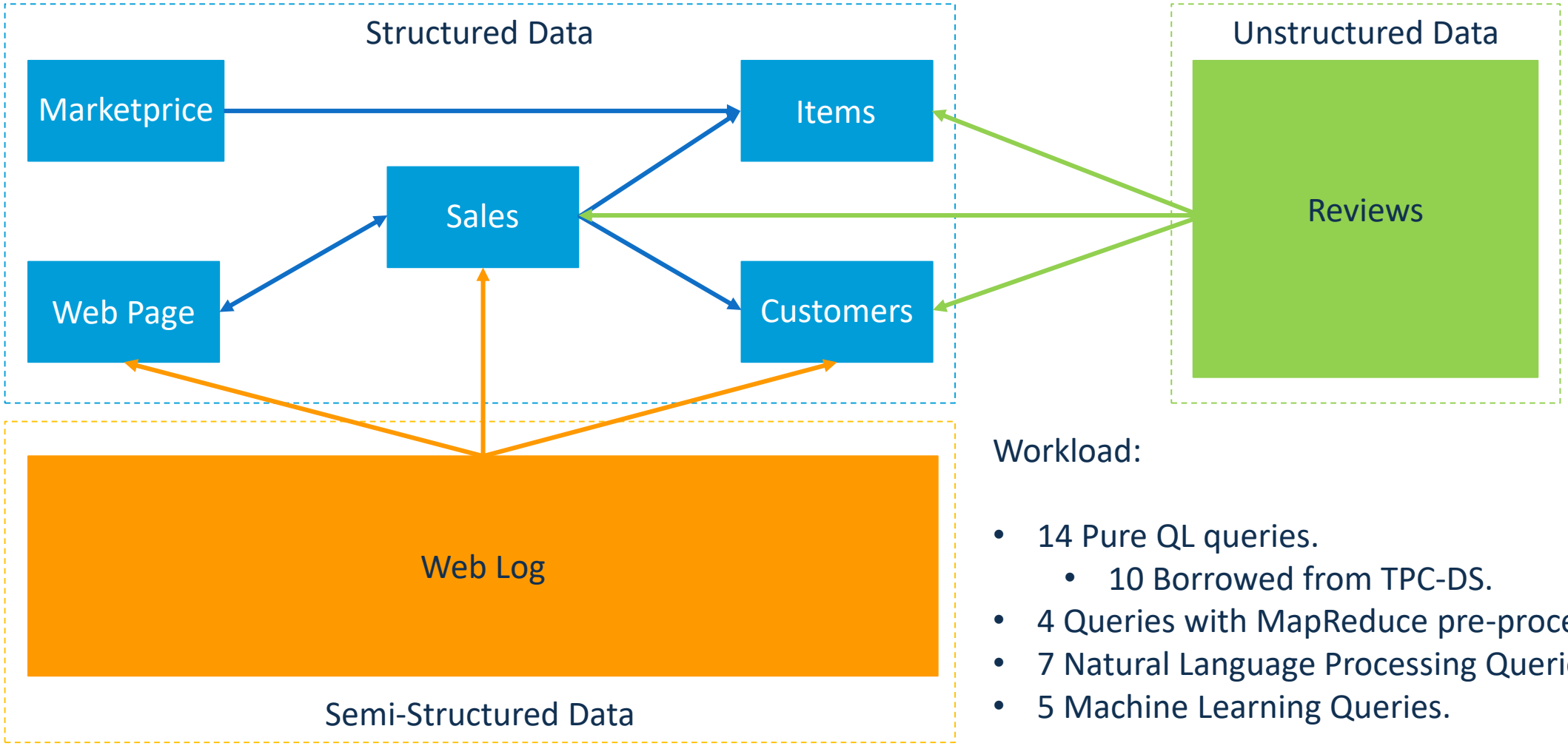
aloja

What is BigBench (TPCx-BB[1])?

- Specification-based benchmark with an open-source implementation, proposed as the first Big Data benchmark standard.
- BigBench covers all major Big Data characteristics
  - Volume -> Scale factor.
  - Velocity -> Table refresh.
  - Variety -> Data type disparity.
- Extension of TPC-DS.
  - Borrows 10 pure QL queries from TCP-DS.
  - Added BigData tables and use cases: Machine Learning, Natural Language Processing, ...
- Can support multiple implementations, multiple BigData engines and table formats.
- Can execute multiple parallel streams.
- Defines scale factors for data.
  - Tested: 100 GB.

[1]: http://www.tpc.org/tpc_documents_current_versions/pdf/tpcx-bb_v1.2.0.pdf

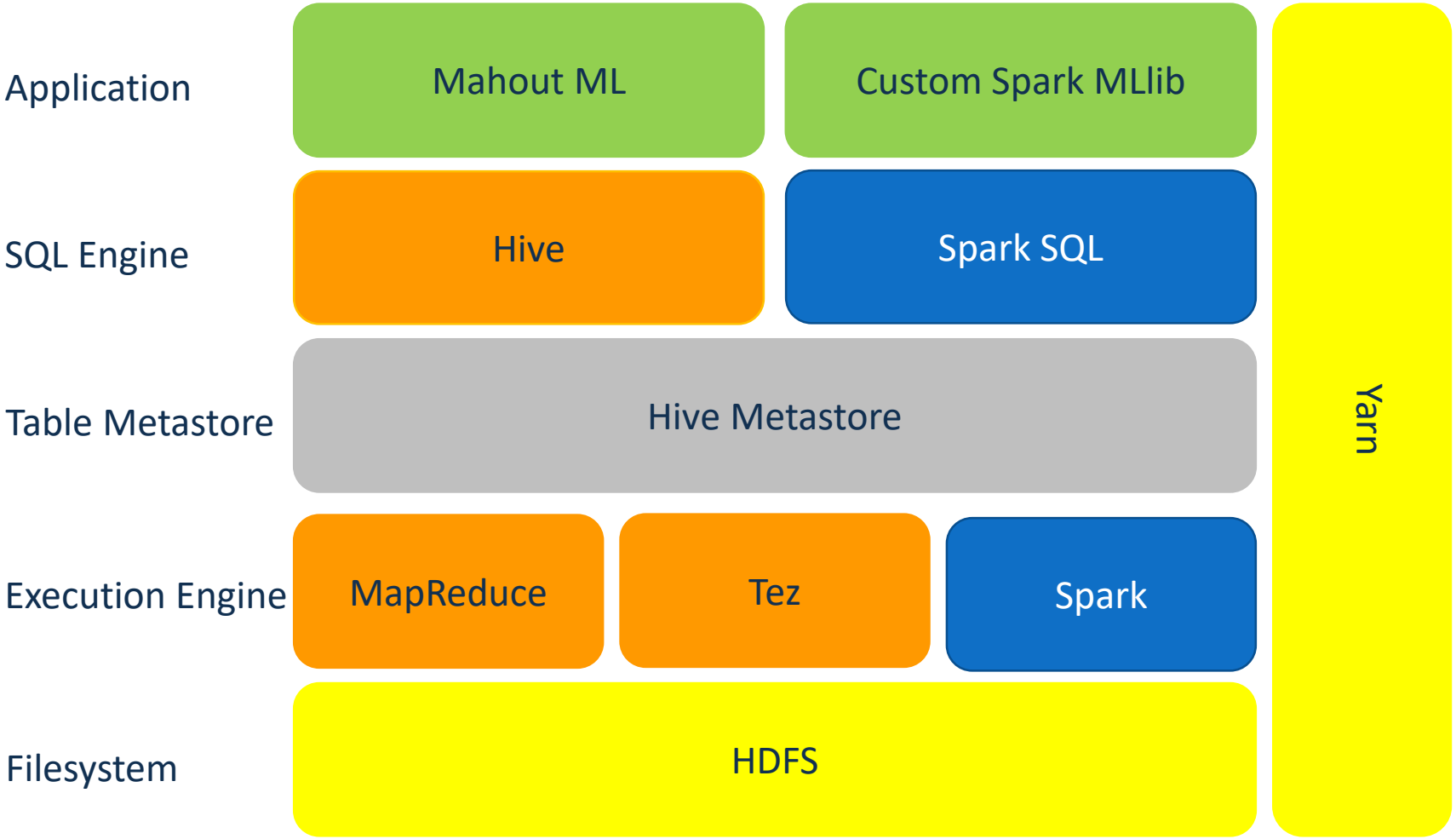
BigBench – Overview



Workload:

- 14 Pure QL queries.
  - 10 Borrowed from TPC-DS.
- 4 Queries with MapReduce pre-processing.
- 7 Natural Language Processing Queries.
- 5 Machine Learning Queries.

BigBench v1.2 – Reference Implementation



Benchmarked systems:

- Hive + MapReduce + Mahout
- Hive + MapReduce + Spark_MLlib
- Hive + Tez + Mahout
- Hive + Tez + Spark_MLlib
- Spark SQL + Mahout
- Spark SQL + Spark_MLlib
- Spark 2 SQL + Mahout

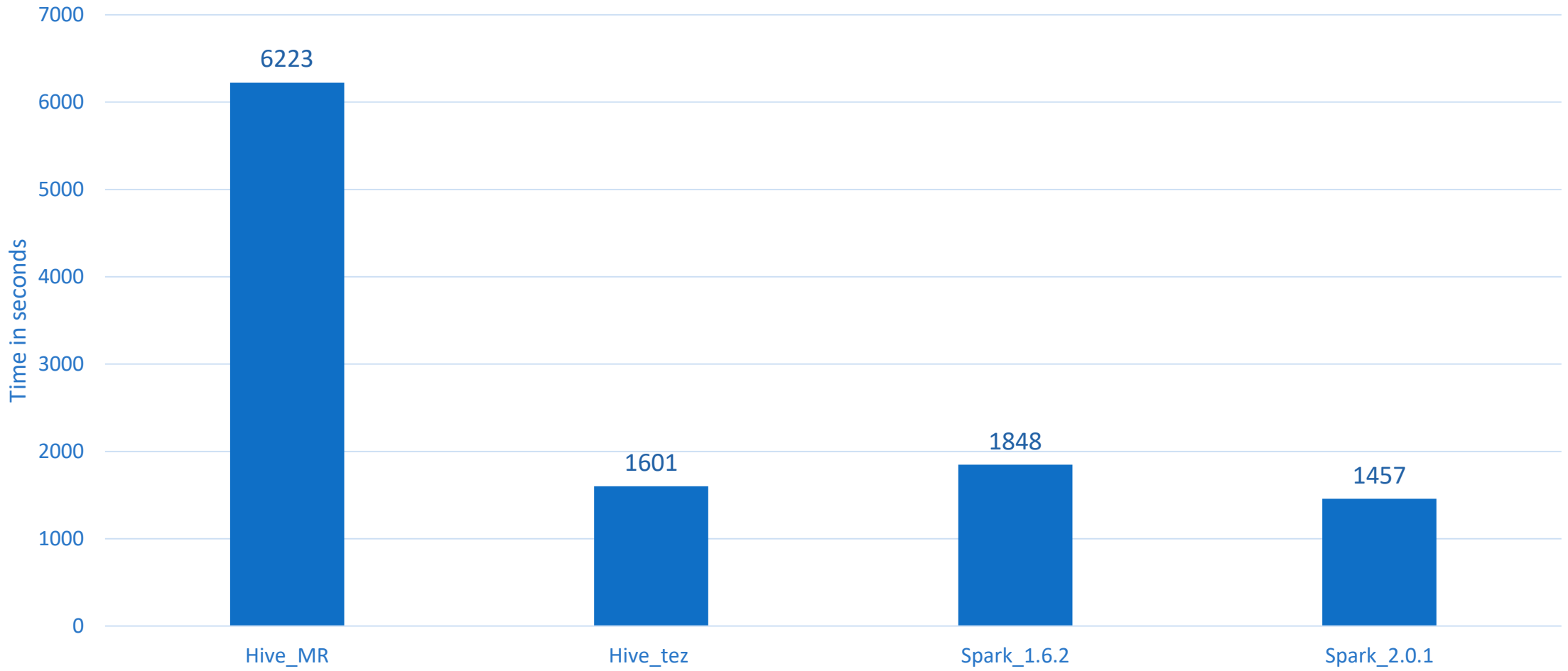
Work in progress:

- Hive 2
- Spark 2 SQL + Spark_MLlib

The cluster – HDInsight PaaS

Model	HDInsight D4v3
# Head nodes	2
# Working nodes	4
# Zookeeper nodes	3
CPU	Intel(R) Xeon(R) CPU E5-2673 v3 8 x 2,4 GHz cores
RAM	28 GB
HDFS	Remote
Software	HortonWorks Data Platform 2.5
Spark config	1 executor/working node 3 cores/executor

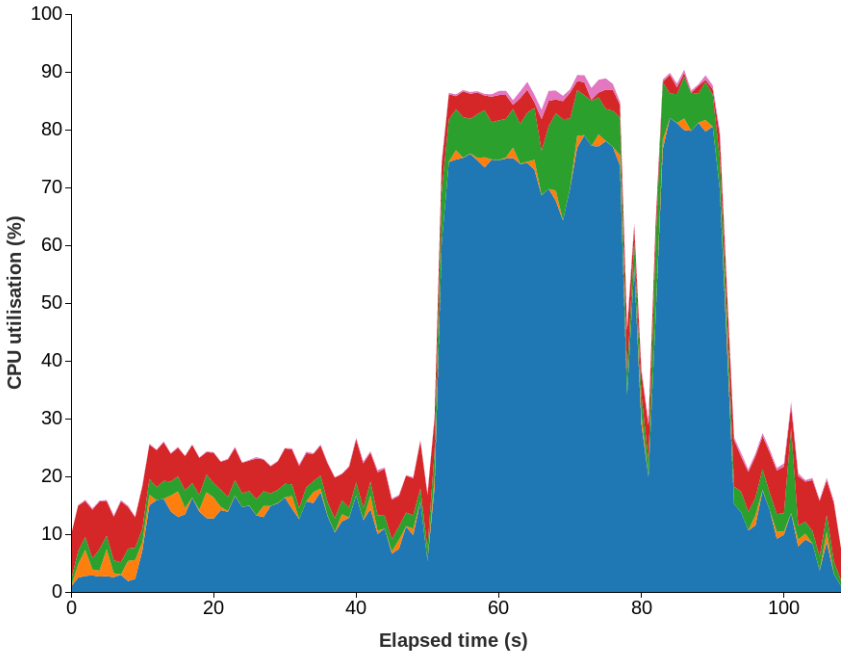
Pure QL



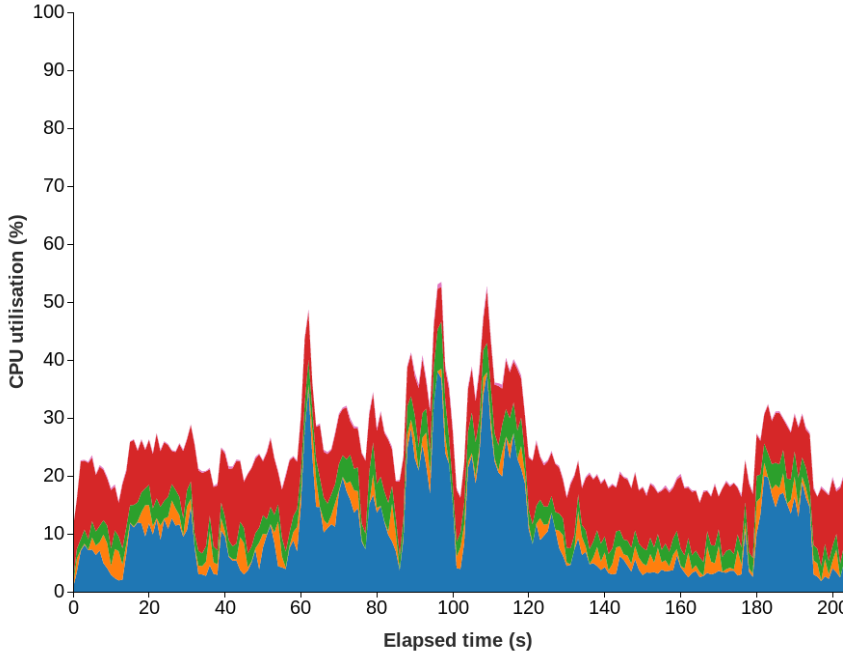
Average of three executions using 100 GB Scale Factor

Query 12 CPU behavior

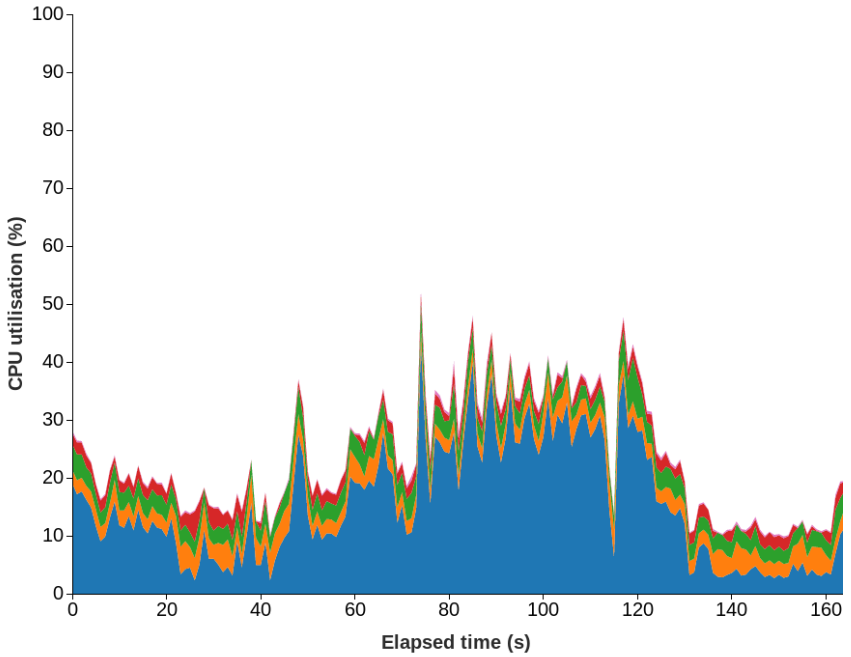
Tez



Spark 1.6.2



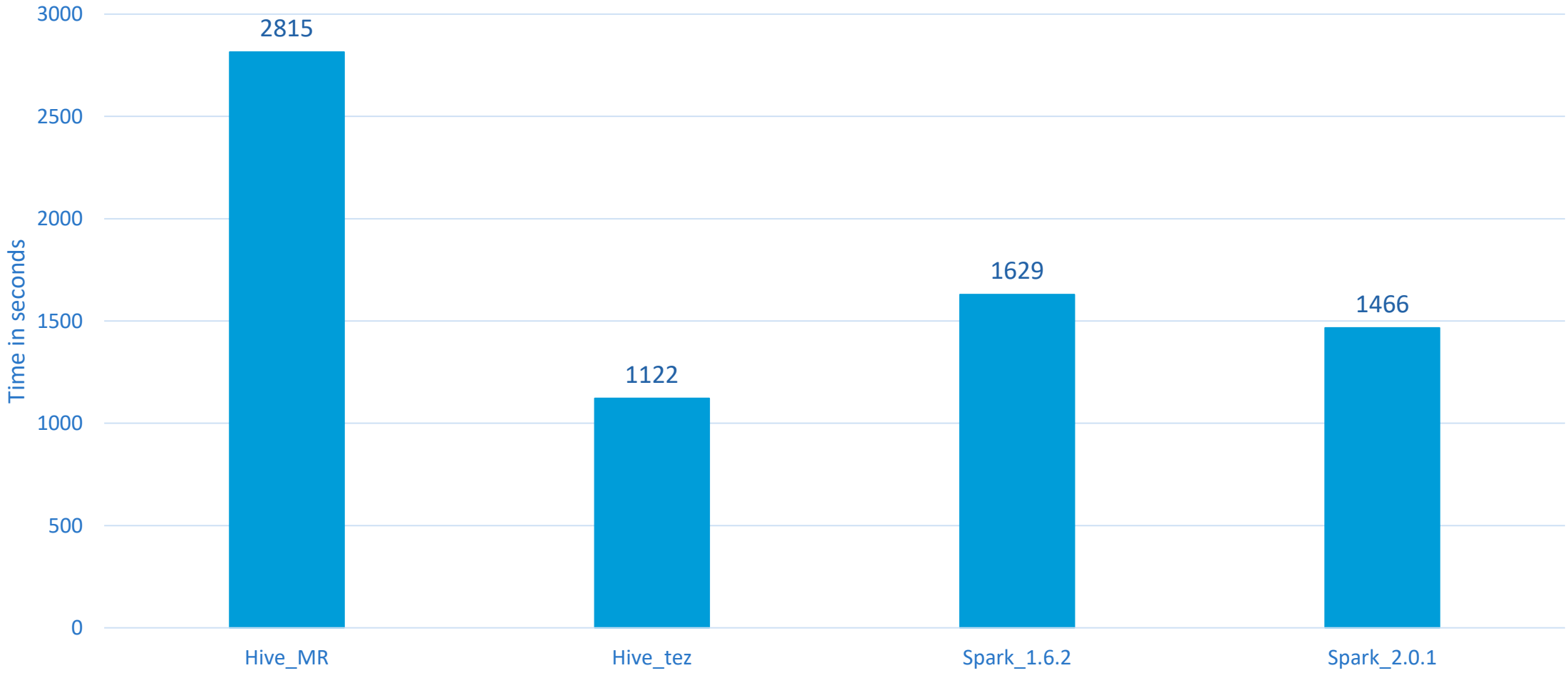
Spark 2.0.2



%usr %nice %sys %iowait %soft

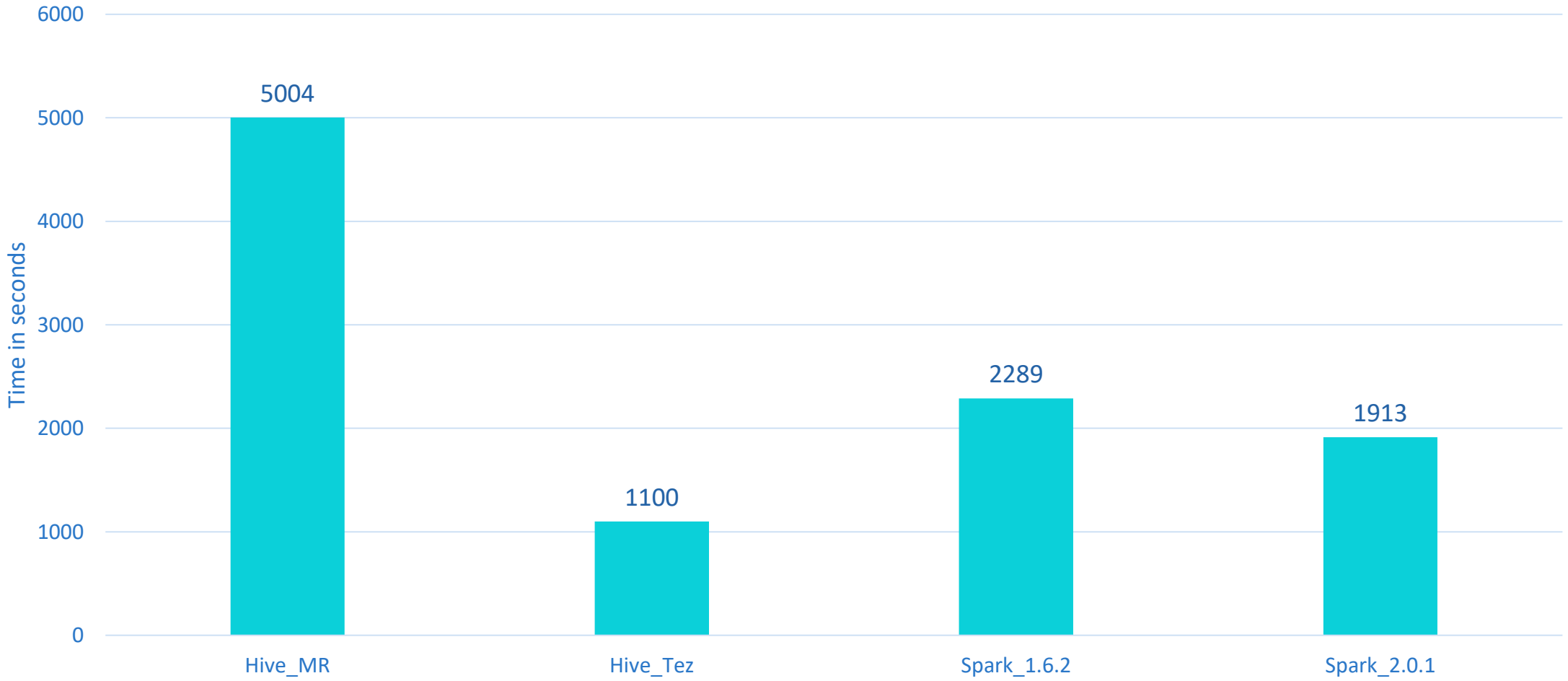
Average of three executions using 100 GB Scale Factor

Custom Reducers



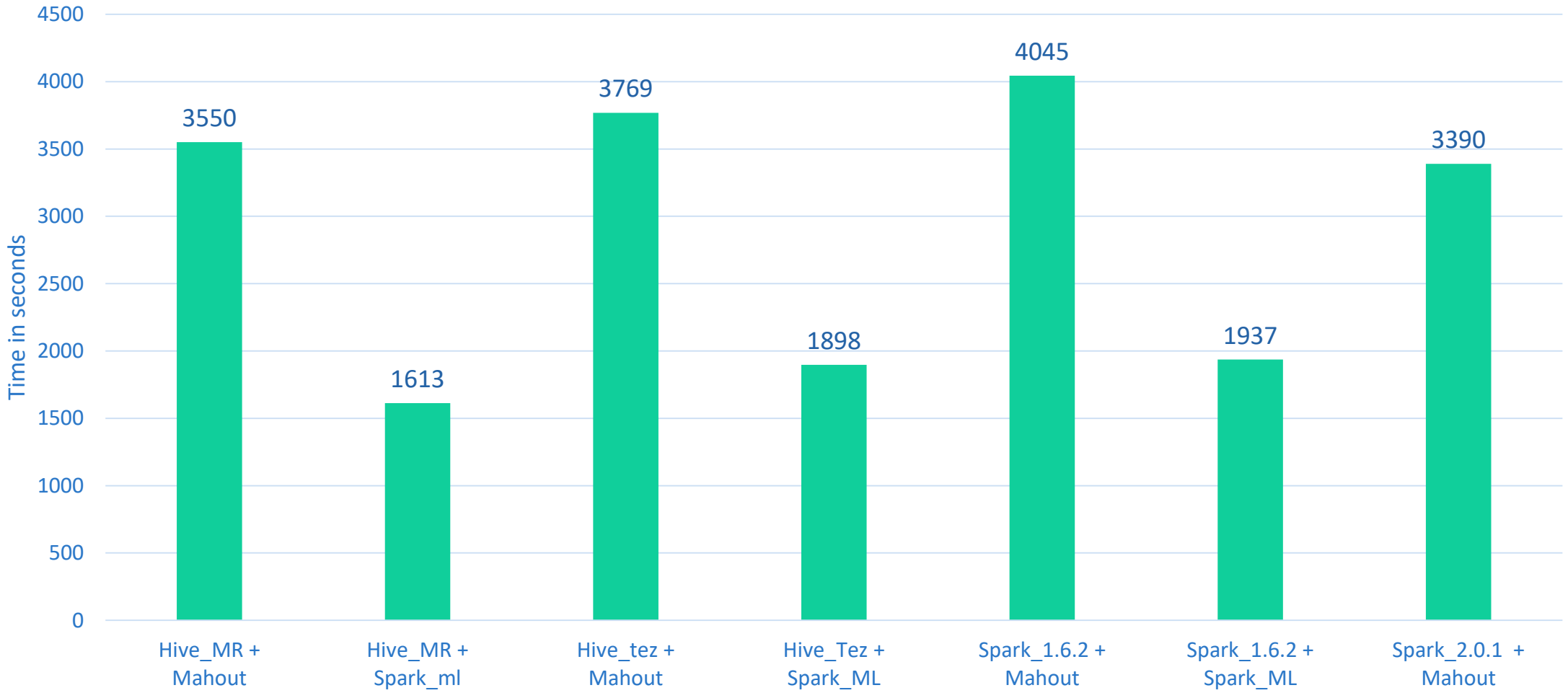
Average of three executions using 100 GB Scale Factor

Natural Language Processing



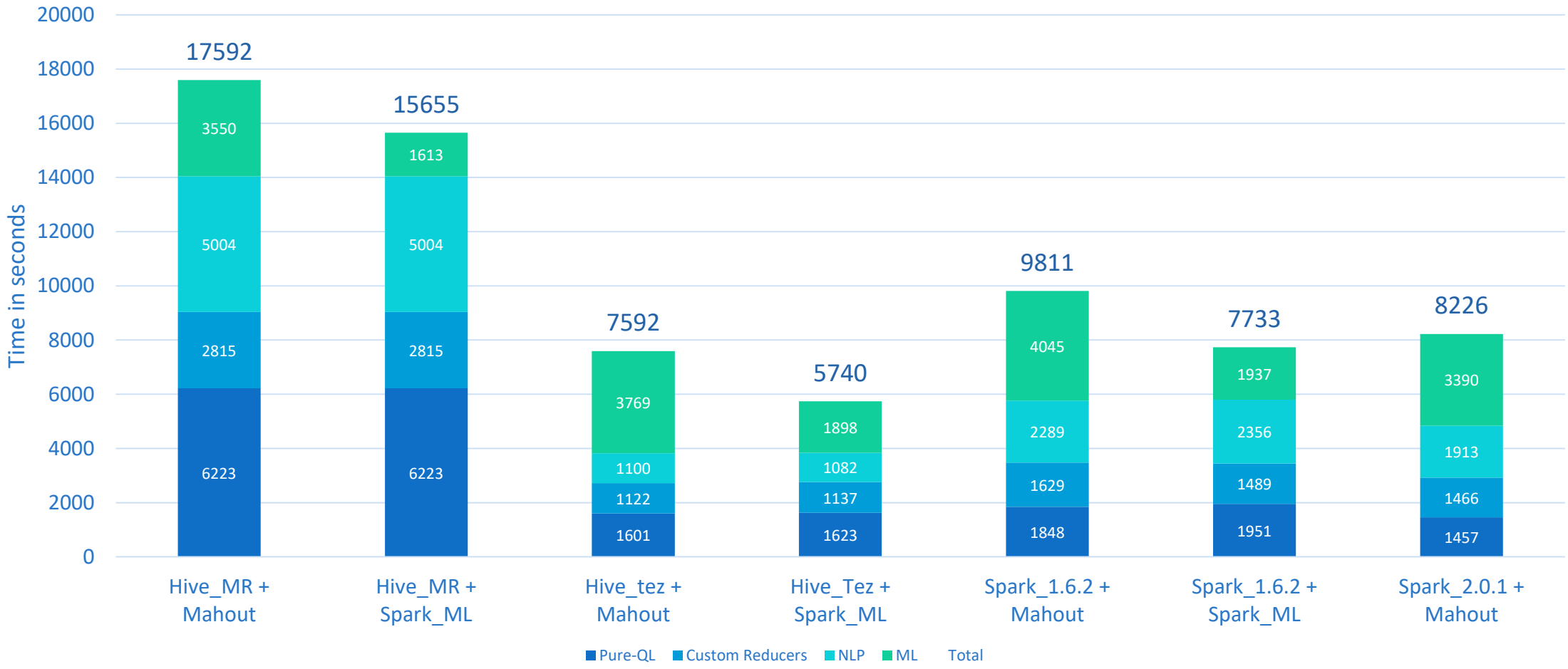
Average of three executions using 100 GB Scale Factor

Machine Learning



Average of three executions using 100 GB Scale Factor

Aggregated Results



Average of three executions using 100 GB Scale Factor

Conclusions

- **Hive on Tez** greatly improves SQL performance over Hive on MapReduce.
  - It is also **faster than Hive on spark 1**.
  - **Hive on spark 2 is slightly faster**.
  - The Spark implementation is based on hive…
- Spark MLlib has an excellent performance over Mahout.
- Best production combination: **Apache Tez for SQL + Spark MLlib for Machine Learning**.

Using BigBench to compare Hive and Spark

Thanks, questions?

Follow up / feedback : Alejandro.montero@bsc.es

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

aloja