Spark Fundamentals

blowUp Data

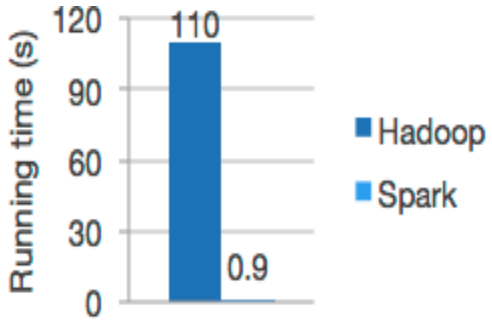
Spark History

- Started by Matei Zaharia at UC Berkeley's AMPLab in 2009
- Open sourced in 2010
- Donated to Apache software foundation and licensed as Apache 2.0
- Now it has more than 1500 contributes and multiple communities
- Many companies are adopting Apache spark to innovate their Big data use cases

blowUp Data

What is Spark

Lightning-fast in-memory computation engine for large-scale data processing



Logistic regression In Hadoop and Spark

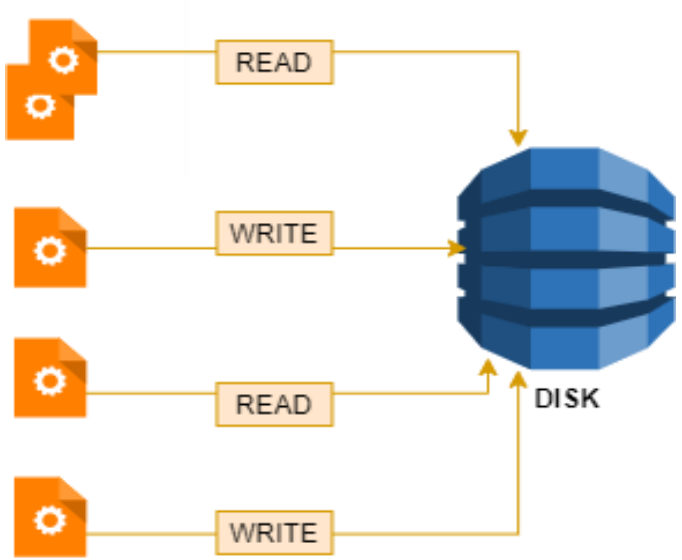
Speed

Run workloads 100x faster.

blowUp Data

What is Spark

Lightning-fast in-memory computation engine for large-scale data processing



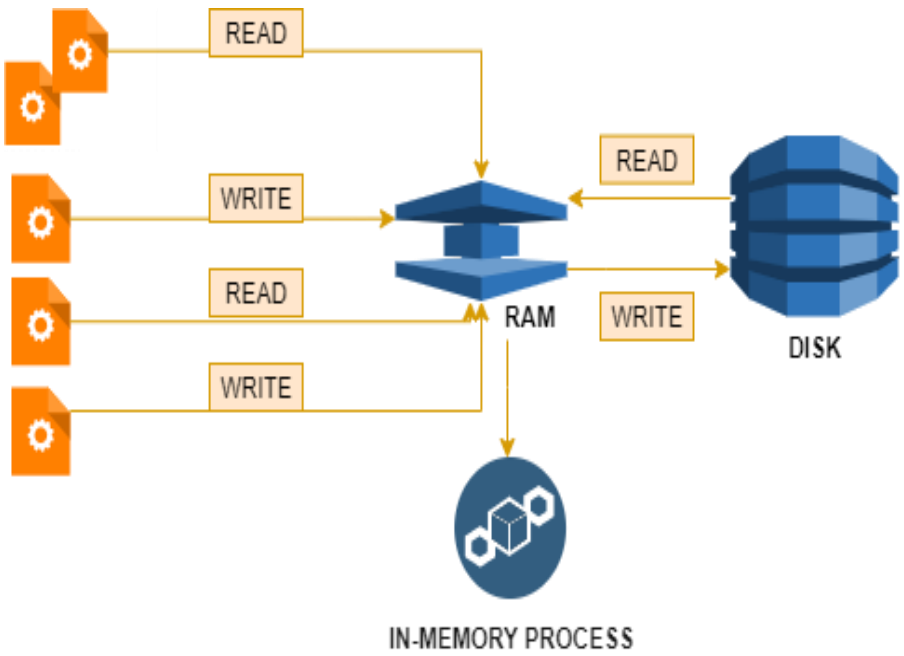
Hadoop Data processing

1. Higher Bandwidth (I/O)
2. Higher Latency
3. External Job scheduler

blowUp Data

What is Spark

Lightning-fast in-memory computation engine for large-scale data processing

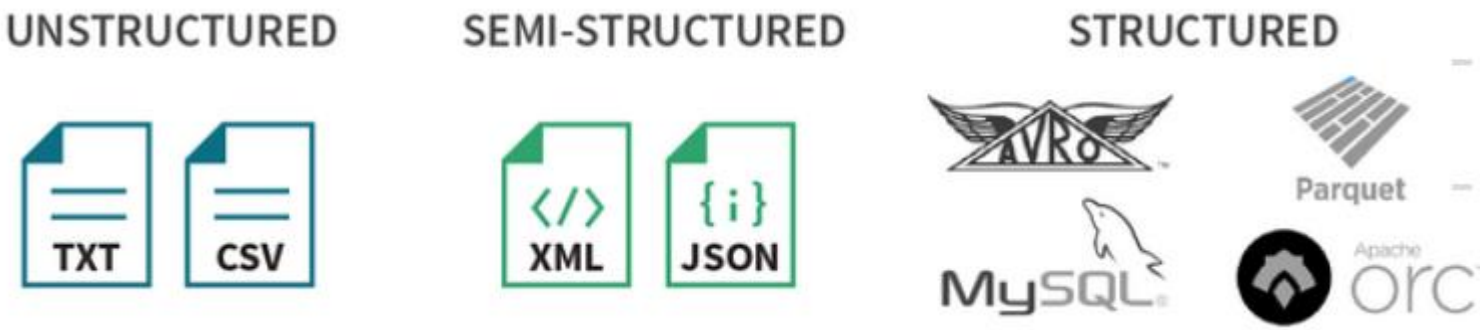


1. Lower Bandwidth (I/O)
2. Lower Latency
3. No External Scheduler

blowUp Data

What is Spark

Lightning-fast in-memory computation engine for large-scale data processing



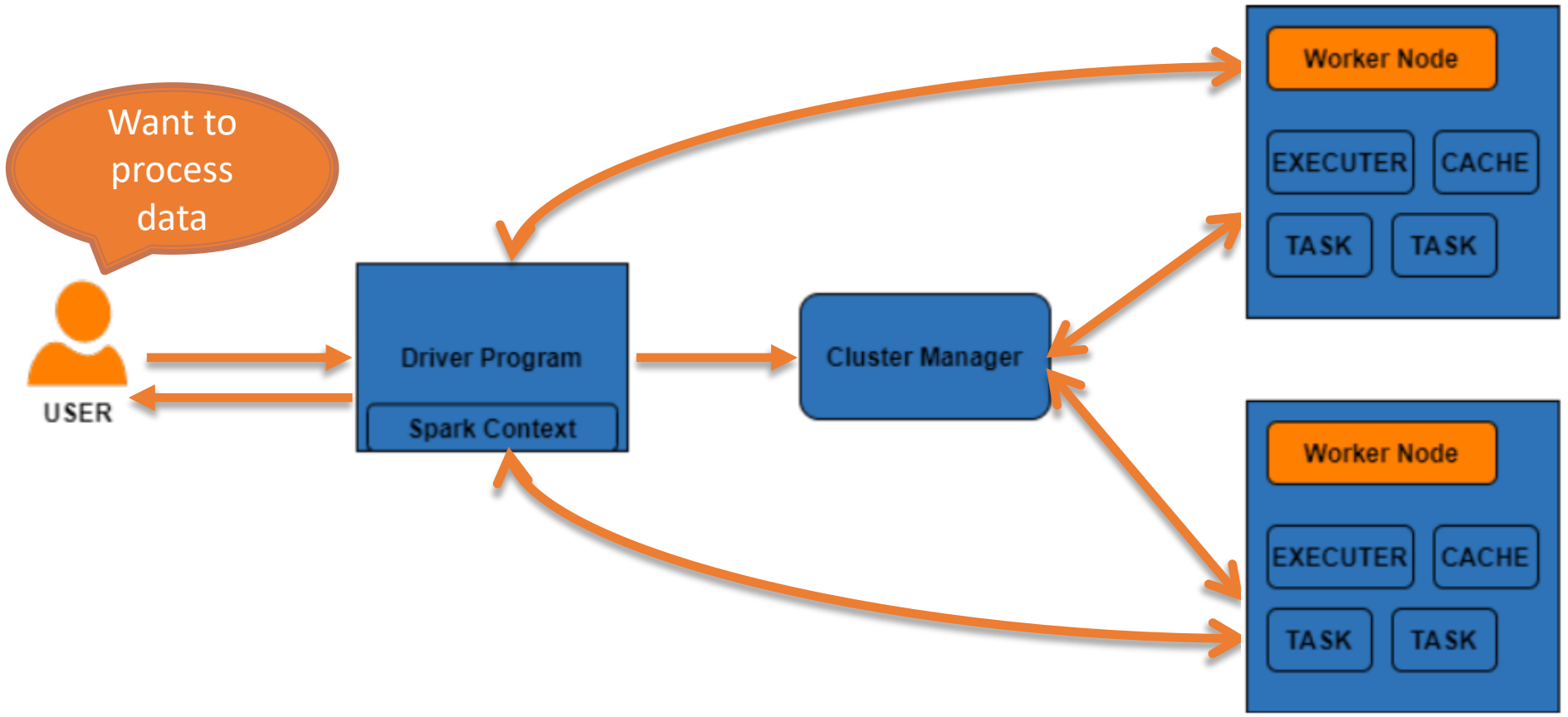
blowUp Data

Spark Components

1. **Spark CORE** — Core engine for large scale parallel and distributed data processing. Spark RDD-Resilient Distributed Dataset
2. **Spark SQL** — Distributed framework for structured data processing. Spark Dataframe and dataset
3. **Spark Streaming** — Scalable, high throughput and fault tolerant processing of streams of data. Dstreams
4. **Spark ML** — Scalable Machine learning library for various algorithm with high speed
5. **Spark GraphX** — Parallel processing engine for network graphs and data store

blowUp Data

Spark Architecture



blowUp Data

- Driver program:
  - Creates the Spark Context object. It coordinate the spark applications.
  - Negotiate the resources with cluster manager
  - Sends application code to executors like jar, python files
  - Spark context sends task to executors to run
- Cluster manager:
  - Allocates resources across the application
  - It can be either Hadoop yarn, apache mesos, spark standalone, Kubernetes
- Worker node:
  - Slave node to run the application code in cluster
- Executors:
  - Runs the task and keeps the data in memory or disk
- Task:
  - Unit of task given to executors for running

blowUp Data

Happy Learning see you again☺

blowUp Data