Week 2, Wednesday, May 4

**Apache Spark:**

The most used and popular engine for distributed computing. Used for analytics (like, mashing learning) and data processing (like, etl, elt)

Work is distributed across multiple systems to get the job done

\*\*\*\*\* **map- reduce** uses hadoop a java virtual machine

Support sql

\*\*\*\*\* **spark** uses JVM,

support Java/Scala, python, R, SQL languages

interactive development, you run a line of code and can see the results instantly

**spark architecture:**

1. Spark core, responsible to run code across the whole system. Used for unstructured and semi structured data, this is where we convert data from unstructured to semi and then to structured data
2. Spark SQL structured data,

Unstructured data: not parsed, no data type

Semi structured data: parsed, no data type and are categorized

Structured data: have a schema with data types (rows & columns and each columns has data type)

1. Spark streaming: real time data processing (like credit card fraud check), near real time in seconds
2. Mlib Machine Learning: forecast
3. graph x processing:
4. Standalone Scheduler: How you deployed your spark
5. YARN:
6. Mesos:

**Skill required for spark:**

* Functional programing
* Scala fundamental, Scala functional aspects
* JVM fundamentals
* JAVA
* ……..

**Challenges of Big data:**

* Operational Challenges:
  + big data processing (the data growing is too much) and discovery (\*\*very important to understand),
  + Present it to business (),
  + hardware and network Failures,
* DATA challenges:
  + Volume
  + Velocity
  + Variety of data
* Computation challenges
  + CPU
  + RAM
  + NETWORK
  + FAILURES

**Graphical user interface, application

Description automatically generated**

**Text

Description automatically generated**

Diagram

Description automatically generated

MAPPING: bring unstructured data to structured data

Shuffling: co-grouping, moving up data from one system to another system, its expensive

Reducing: count and aggregate

Final result: collect all the results Diagram

Description automatically generated

In hadoop:

Data nodes are store in hdfs

In spark:

Data nodes are store in RAM

Diagram

Description automatically generated with low confidence

If there’s a failure it get pack to replication

Diagram

Description automatically generated with low confidence

Chart, diagram

Description automatically generated

SPARK provide in memory computing

Diagram

Description automatically generated

Spark doesn’t not have replication: if there’s a failure spark do recomputations(not the whole data but the computation technically happened in a what we called failed chunk)

Table

Description automatically generated

Table

Description automatically generated

* Operations and caching??
* Batch: scheduled operations
  + MapReduce has only one execution model
  + Spark have three execution models: batch, streaming (real time), interactive(?)
* Programming environment: see above!

Diagram

Description automatically generated

Outputs are called partitions: part 0, part 1

Diagram

Description automatically generated

Sort, join and merge mean shuffling (need to check this info)

Diagram

Description automatically generated

Diagram

Description automatically generated

Spark development start when they realized MapReduce isn’t good enough

Graphical user interface, text, application

Description automatically generated

Diagram

Description automatically generated

RDD:

[What is a Resilient Distributed Dataset](https://databricks.com/glossary/what-is-rdd)

Chart, bar chart

Description automatically generated

Diagram

Description automatically generated

# 1- create rdd

# 2- tranform

# 3- perform actions

# 4- results

Check S002- SparkBasic

Diagram

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Diagram

Description automatically generated