Yandex

Introduction to Spark

Week outline

- Lesson 1 – Basic concepts
  - RDDs, transformations, actions, resiliency
- Lesson 2 – Advanced topics
  - execution, persistence & caching, broadcast variables, accumulators
- Lesson 3 – Spark and Python
  - integration, examples

Historical background

- 2009 – project started at UC Berkeley's AMPLab
- 2012 – first release (0.5)
- 2014 – became top-level Apache project
- 2014 – reached 1.0
- 2015 – reached 1.5
- 2016 – reached 2.0

First epoch (2009-2012)

- Key observations
  - Underutilization of cluster memory
    - for many companies data can fit into memory either now, or soon
    - memory prices were decreasing year-over-year at that time
  - Redundant disk I/O
    - especially in iterative MR jobs
- Lack of higher-level primitives in MR
    - one has to redo joins again and again
    - one has to carefully tune the algorithm

First epoch (2009-2012)

- Key observations
  - Underutilization of cluster memory
  - Redundant disk I/O
  - Lack of higher-level primitives in MR

- Key outcomes
  - RDD abstraction with rich API
  - In-memory distributed computation platform

Second epoch (2012-2014)

- Key observations
  - No "one system to rule them all"
    - typical cluster would include a dozen of different systems tailored for specific applications
    - recurrent data copying between the systems increases timings

- Increasing demand for interactive queries and stream processing
  - due to raise of data-driven applications
    - need for fast ad-hoc analytics
    - need for fast decision-making

Second epoch (2012-2014)

- Key observations
  - No "one system to rule them all"
  - Increasing demand for interactive queries and stream processing

- Key outcomes
  - Separation of Spark Core and applications on top of the core:
    - Spark SQL
    - Spark Streaming
    - Spark GraphX
    - Spark MLlib

Third epoch (2014-now)

- Key observations
  - Increasing use of machine learning
  - Increasing demand for integration with other software (Python, R, Julia…)

Third epoch (2014-now)

- Key observations
  - Increasing use of machine learning
  - Increasing demand for integration with other software (Python, R, Julia…)

- Key outcomes
  - Focus on ease-of-use
  - Spark Dataframes as first-class citizens

This week

› Focus on fundamentals (Spark Core)

› Apply for other courses in the specialization to learn more
  › about data warehousing & analytics
  › about machine learning
  › about real-time applications

BigDATAteam