Course Title

Big Data Emerging Technologies

❖ **Modules**

1. Big Data Rankings & Products
2. Big Data & Hadoop
3. Spark
4. Spark ML & Streaming
5. Storm
6. IBM SPSS Statistics Project

Big Data

Apache Spark

Apache Spark

❖ Spark

- Spark is a Big Data general processing technology
- Spark is currently the most popular and most active open source big data project in the world
  - Before Spark, Hadoop was the most widely used open source big data technology

Apache Spark

❖ Spark Technical Characteristics

- Spark conducts its own cluster management
  - Spark supports batch applications, iterative algorithms, interactive queries
  - Spark is independent of Hadoop
  - Spark can use Hadoop for Storage or Processing

Apache Spark

❖ Spark Technical Characteristics

- Spark has a built in MLlib (Machine Learning library)
- Spark supports Stream Processing functionality
  - Spark has a streaming mode for real-time applications, which uses micro-batch technology
  - Spark has been reported to be tens to hundreds of times faster than Hadoop

Apache Spark

❖ Spark Technical Characteristics

- Spark is very fast and uses improved data processing techniques
  - In-memory (RAM) processing
  - RDD (Resilient Distributed Datasets)
  - DAG (Directed Acyclic Graph)
  - Advanced Scheduling
  - Persisting techniques
  - Real-time Streaming
  - etc.

Apache Spark

❖ **Spark**

- **Spark does not have its own unique distributed storage system, but is built to use various third-party distributed file organizing systems**
- **Many Spark systems are connected to Hadoop systems**

Apache Spark

❖ **Spark**

- **Hadoop's MapReduce is replaced with Spark's RDD (Resilient Distributed Datasets) and DAG, Transformations, and Actions**
- **Spark uses the HDFS (Hadoop Distributed File System) through the YARN resource manager**

Apache Spark

❖ **Spark**

- **Spark's advanced analytics applications and built-in ML (Machine Learning) library functions enable remarkable information extraction from data stored in HDFSs and various datasets**
- **Hadoop requires a 3rd party ML library Mahout for ML functions**

Apache Spark

❖ **Spark compared to Hadoop**

- **Hadoop was slow because MapReduce saves all of its processed data in its physical storage medium (commonly HDDs) after each operation, to be fault tolerant (resilient from crashes)**
- **Hadoop repeats this process multiple times in a Job, which makes it even slower**

Apache Spark

❖ **Spark Applications**

- **Retailer recommendation engines**
- **Industry machinery and manufacturing monitoring & automation**
- **Prediction systems that estimate when parts will malfunction, when best to replace, and when to order replacement components**
- **Controllers for IoT (Internet of things) & CPS (Cyber Physical Systems)**

Apache Spark

❖ **Evolution of Spark**

- **Spark and Mesos were developed by the AMPLab (Algorithms Machines People Lab) at UC Berkeley**
- **In 2010, Spark became an Open Source Software based on a BSD (Berkeley Software Distribution) license**

Apache Spark

❖ Evolution of Spark

- In 2013, Spark was donated to the Apache software foundation
- In 2014, Apache Spark became a top-level Apache project
- In 2014, May 30, Apache Spark was initially released

Apache Spark

❖ Spark Characteristics

- Spark scales very well
  - Spark can be executed on clusters consisting of thousands of nodes processing petabyte (10^{24} Bytes) size databases

Apache Spark

❖ Spark & Hadoop Relation

- Hadoop and Spark are both big data technologies
- Both provide some of the most popular big data tools
- Both are Apache Software Foundation tools
- Hadoop and Spark systems can work together

Apache Spark

❖ Spark & Hadoop Relation

- Many Spark systems are connected to a Hadoop HDFS through YARN
- Both are scalable and more data drives can be added to the network as the dataset grows
- Task management and data processing schemes are different

Big Data

References

References

- Holden Karau, Andy Konwinski, Patrick Wendell, and Matei Zaharia, *Learning Spark: Lightning-Fast Big Data Analysis*. 1st Edition. O'Reilly, 2015.
- Sameer Farooqui, Databricks, **Advanced Apache Spark Training**, Devops Advanced Class, Spark Summit East 2015, http://slideshare.net/databricks, www.linkedin.com/in/blueplastic, March 2015.
- Apache Spark documents (all documents and tutorials were used)
  - http://spark.apache.org/docs/latest/rdd-programming-guide.html
  - http://spark.apache.org/docs/latest/rdd-programming-guide.html#working-with-key-value-pairs
  - https://spark.apache.org/docs/2.2.0/rdd-programming-guide.html#rdd-persistence
- Wikipedia, www.wikipedia.org
- Stackoverflow, https://stackoverflow.com/questions
- Bernard Marr, "Spark Or Hadoop -- Which Is The Best Big Data Framework?," Forbes, Tech, June 22, 2015.
- Quick introduction to Apache Spark, https://www.youtube.com/watch?v=TgiBvKcGL24
- Wide vs Narrow Dependencies, https://github.com/rohgar/scala-spark-4/wiki/Wide-vs-Narrow-Dependencies

References

- Partitions and Partitioning, https://jaceklaskowski.gitbooks.io/mastering-apache-spark/spark-rdd-partitions.html
- Neo4j, "From Relational to Neo4j," https://neo4j.com/developer/graph-db-vs-rdbms/ (last accessed Jan. 1, 2018).

Image Sources

- By Robivy64 at English Wikipedia [Public domain], via Wikimedia Commons
- Teravolt at English Wikipedia [CC BY 3.0 (http://creativecommons.org/licenses/by/3.0)], via Wikimedia Commons
- By Konradr (Own work) [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/)], via Wikimedia Commons