Big Data

Spark SQL

Spark SQL

❖ **Spark SQL**

- **Supports SQL operations based on the Spark Core**
- **Can easily include SQL queries into Spark programs (Java, Scala, Python, R)**

Spark SQL

❖ **Spark SQL**

- **Programmers can connect to any data source (Hive, Avro, Parquet, ORC, JSON, JDBC) the same way and easily join datasets across these different data source types**

Spark SQL

❖ **Spark SQL**

- **SchemaRDD based data abstraction**
- **Provides data abstraction based on DataFrames**
- **Provides support for structured and semi-structured data**

Spark SQL

❖ Spark SQL

- Provides DSL (Domain-Specific Language, in Scala, Java, or Python) to manipulate DataFrames
- BI (Business Intelligence) tools can be easily used as the Server mode provides industry standard JDBC and ODBC connectivity

Spark SQL

❖ Spark SQL

- Programmers can connect to any data source (Hive, Avro, Parquet, ORC, JSON, JDBC) the same way and easily join datasets across these different data source types

Spark SQL

❖ **Dataset**

- **Distributed collection of data**
- **Interface of Spark that provides RDD and Spark SQL functionality**
- **Constructed from JVM objects and processed using Transformations**

Spark SQL

❖ **DataFrame**

- **Dataset organized into named columns**
- **Similar to a R or Python based Data Frame, or RDB (Relational Database) Table**

Spark SQL

❖ DataFrame

- **DataFrames are created from**
  - **Structured data files**
  - **Tables in Hive**
  - **External databases**
  - **Existing RDDs**

Spark SQL

❖ Data abstraction

- **Reducing process to obtain a simplified representation of the whole dataset**

Spark SQL

❖ **ODBC (Open Database Connectivity)**

- **Standard API (Application Programming Interface) to access DBMSs (Database Management Systems)**
- **API designed to be independent of the DBS (Database System) and OS (Operating System)**

Spark SQL

❖ **JDBC (Java Database Connectivity)**

- **Java API used for Client access to Databases**
- **Java Standard Edition platform based API**
- **Enables database Query and Update functionality**

Big Data

Spark GraphX

Spark GraphX

❖ **GraphX**

- **Distributed graph-processing framework**
- **Originally developed by UC Berkeley's AMPLab and Databricks**
- **Later donated to the Apache Software Foundation and the Spark project**

Spark GraphX

❖ **GraphX**

- **Examines the entire graph (batch process) to derive the optimal solution/answer**
- **Graph computation API (Application Programming Interface)**
- **Optimized runtime processing of user-defined graphs**

Spark GraphX

❖ **GraphX**

- **Spark's distributed graph processing framework is based on RDDs**
  - **Since RDDs are immutable, GraphX's graphs are immutable**
  - **GraphX is unsuitable for database graphs that need to be updated**
    - **So if changes are made to GraphX's values or structure, a new graph will be created**
    - **Some programmers use GDB (Graph Database) technology with Spark GraphX**

Spark GraphX

❖ GraphX

- **Spark's distributed graph processing framework is based on RDDs**
  - **Entire graph of the RDD is transformed into a combination of VertexRDDs and EdgeRDDs**
  - **VertexRDDs and EdgeRDDs are suitable for graph computation and easy to apply graph optimization algorithms**

Spark GraphX

❖ GraphX

- **Spark's distributed graph processing framework is based on RDDs**
  - **APIs enable implementation of massively parallel algorithms**
  - **MapReduce style API**
  - **Pregel API**

Spark GraphX

❖ **GraphX**

- **Pregel API**
  - **Enables message sending computation based on**
    - Vertex attributes
    - Constrains messages to the graph structure
  - **Enables more efficient distributed execution and flexibility in graph-based computations**
  - **Provides database abstraction in graph form**

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