#Comparison #DevOps Tools

Comparing Apache Hive vs. Spark

Daniel Berman
Aug 5th, 2019

Introduction

Hive and Spark are two very popular and successful products for processing large-scale data sets. In other words, they do big data analytics. This article focuses on describing the history and various

What is Hive?

More on the subject:

[What's New in Logz.io - September 2019]

[Web Server Monitoring Your Application on Nginx with Logz.io]

[Formatting Fields]

Hive is an open-source distributed data warehousing database which operates on Hadoop Distributed File System. Hive was built for querying and analyzing big data. The data is stored in the form of tables (just like RDBMS). Data operations can be performed using a SQL interface called HiveQL. Hive brings in SQL capability on top of Hadoop, making it a horizontally scalable database and a great choice for DWH environments.

A Bit of Hive's History

Hive (which later became Apache) was initially developed by Facebook when they found their data growing exponentially from GBs to TBs in a matter of days. At the time, Facebook loaded their data into RDBMS databases using Python. Performance and scalability quickly became issues for them, since RDBMS databases can only scale vertically. They needed a database that could scale horizontally and handle really large volumes of data. Hadoop was already popular by then; shortly afterward, Hive, which was built on top of Hadoop, came along. Hive is similar to an RDBMS database, but it is not a complete RDBMS.

Why Hive?

HiveQL, makes it easier for developers who have RDBMS backgrounds to build and develop faster performing, scalable data warehousing type frameworks.

Hive Features and Capabilities

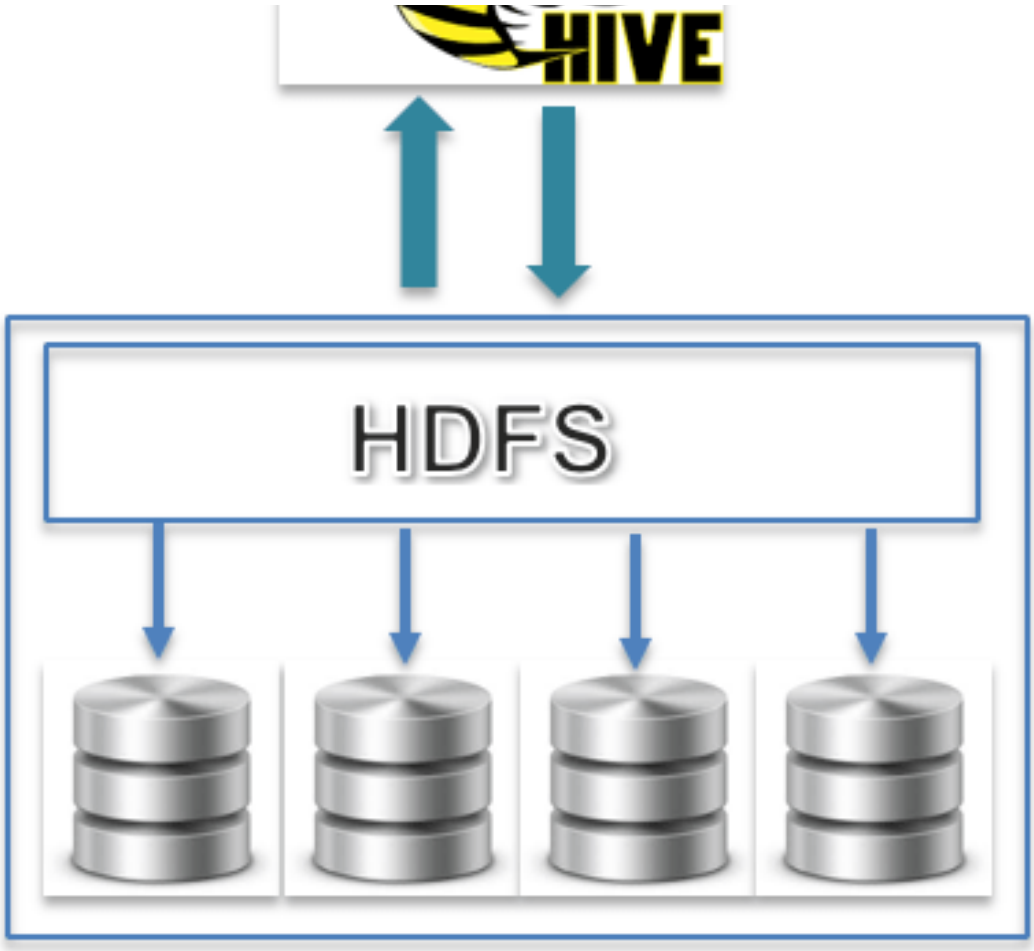
Hive comes with enterprise-grade features and capabilities which can help organizations build efficient, high-end data warehousing solutions.

Some of these features include:

- Hive uses Hadoop as its storage engine and only runs on HDFS.
- It is specially built for data warehousing operations and is not an option for OLTP or OLAP.
- HiveQL is an SQL engine which helps build complex SQL queries for data warehousing type operations. Hive can be integrated with other distributed databases like HBase and with NoSQL databases like Cassandra

Hive Architecture

Hive Architecture is quite simple. It has a Hive interface and uses HDFS to store the data across multiple servers for distributed data processing.



Hive for Data Warehousing Systems

Hive is a specially built database for data warehousing operations, especially those that process terabytes or petabytes of data. It is an RDBMS-like database, but is not 100% RDBMS. As mentioned earlier, it is a database which scales horizontally and leverages Hadoop's capabilities, making it a fast-performing, high-scale database. It can run on thousands of nodes and can make use of commodity hardware. This makes Hive a cost-effective product that renders high performance and scalability.

Hive Integration Capabilities

such as Spark, Kafka and Flume.

Hive's Limitations

Hive is a pure data warehousing database which stores data in the form of tables. As a result, it can only process structured data read and written using SQL queries. Hive is not an option for unstructured data. In addition, Hive is not an ideal for OLTP or OLAP kinds of operations.

What is Spark?

Spark is a distributed big data framework which helps extract and process large volumes of data in RDD format for analytical purposes. In short, it is not a database, but rather a framework which can access external distributed data sets using RDD (Resilient Distributed Data) methodology from data stores like Hive, Hadoop, and HBase. Spark operates quickly because it performs complex analytics in-memory.

What Is Spark Streaming?

Spark streaming is an extension of Spark which can stream live data in real-time from web sources to create various analytics. Though there are other tools, such as Kafka and Flume, that do this, Spark becomes a good option performing really complex data analytics is necessary. Spark has its own SQL engine and works well when integrated with Kafka and Flume.

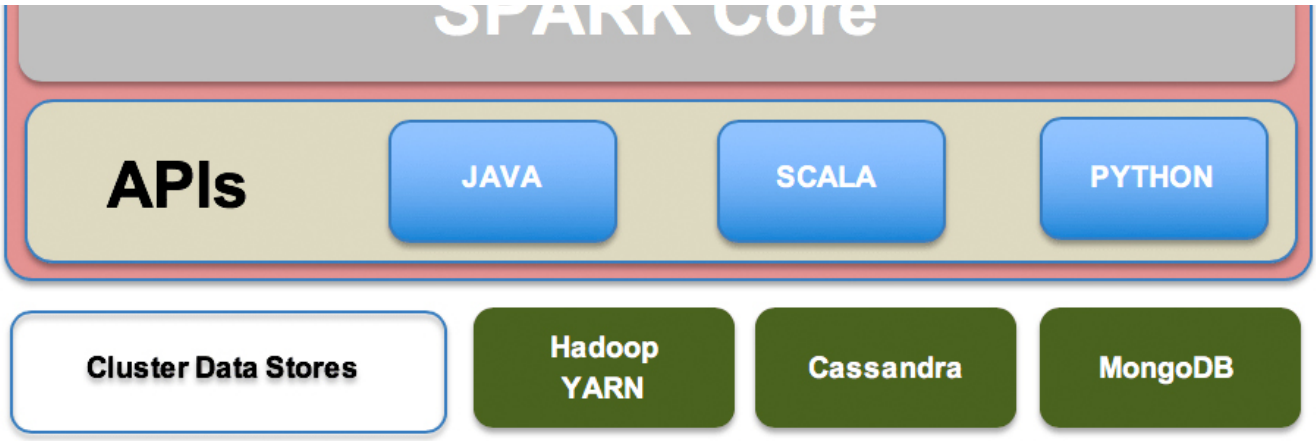
A Bit of Spark's History

Why Spark?

The core strength of Spark is its ability to perform complex in-memory analytics and stream data sizing up to petabytes, making it more efficient and faster than MapReduce. Spark can pull the data from any data store running on Hadoop and perform complex analytics in-memory and in parallel. This capability reduces Disk I/O and network contention, making it ten times or even a hundred times faster. Also, data analytics frameworks in Spark can be built using Java, Scala, Python, R, or even SQLs.

Spark Architecture

Spark Architecture can vary depending on the requirements. Typically, Spark architecture includes Spark Streaming, Spark SQL, a machine learning library, graph processing, a Spark core engine, and data stores like HDFS, MongoDB, and Cassandra.



Spark Features and Capabilities

Lightning-fast Analytics

Spark extracts data from Hadoop and performs analytics in-memory. The data is pulled into the memory in parallel and in chunks, then the resulting data sets are pushed across to their destination. The data sets can also reside in the memory until they are consumed.

Spark Streaming

Spark Streaming is an extension of Spark which can live-stream large amounts of data from heavily-used web sources. Because of its ability to perform advanced analytics, Spark stands out when compared to other data streaming tools like Kafka and Flume.

Support for Various APIs

Spark supports different programming languages like Java, Python and Scala which are immensely popular in big data and data analytics spaces. This allows data analytics frameworks to be written in any of these

As mentioned earlier, advanced data analytics often need to be performed on massive data sets. Before Spark came into the picture, these analytics were performed using MapReduce methodology. Spark not only supports MapReduce, it also supports SQL-based data extraction. Applications needing to perform data extraction on huge data sets can employ Spark for faster analytics.

Integration with Data Stores and Tools

Spark can be integrated with various data stores like Hive and HBase running on Hadoop. It can also extract data from NoSQL databases like MongoDB. Spark pulls data from the data stores once, then performs analytics on the extracted data set in-memory, unlike other applications which perform such analytics in the databases.

Spark's extension, Spark Streaming, can integrate smoothly with Kafka and Flume to build efficient and high-performing data pipelines.

Differences Between Hive and Spark

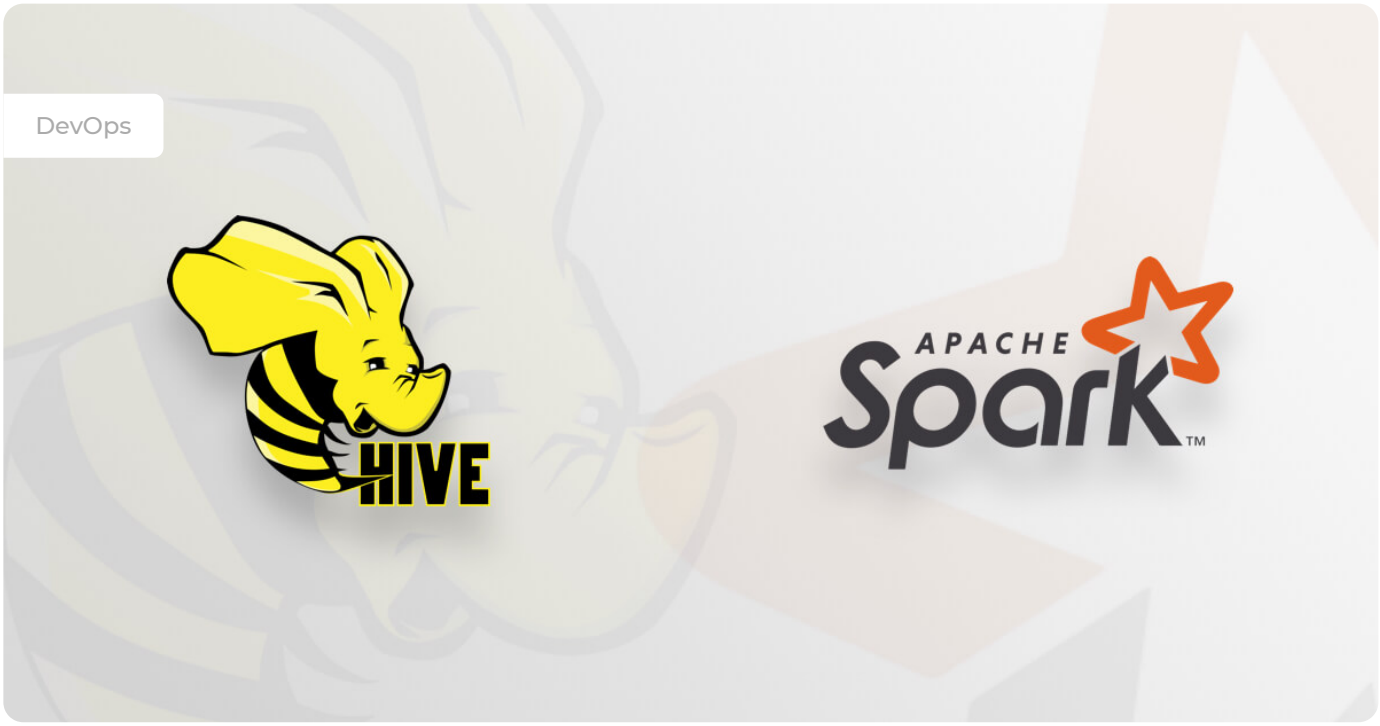
Hive and Spark are different products built for different purposes in the big data space. Hive is a distributed database, and Spark is a framework for data analytics.

Differences in Features and Capabilities

	volumes of data sizing up to petabytes.
Data can be extracted from Hive using its own SQL engine called HiveQL. The data can be extracted only by using SQLs.	Spark performs data analytics using Complex SQLs and also uses the MapReduce mechanism. It supports analytics frameworks written in Java, Scala, and Python.
Hive operates on top of Hadoop.	Spark does not have its own dedicated storage. In fact, it extracts data from external distributed data stores like Hive, HBase running on Hadoop, and MongoDB.
Hive is a data warehousing database.	Spark is best for performing complex and faster in-memory data analytics and live data streaming.
Hive is a best suited for those applications performing DWH operations on RDBMS databases that need a scale-out database.	Spark is best suited for applications performing big data analytics requiring a solution faster than the MapReduce mechanism.

Conclusion

Hive and Spark are both immensely popular tools in the big data world. Hive is the best option for performing data analytics on large volumes of data using SQLs. Spark, on the other hand, is the best option for running big data analytics. It provides a faster, more modern alternative to MapReduce.



Comparing Apache Hive vs. Spark

Keeping it Local: Bringing Logz.io to an AWS Region Near You

Learn

OpenSearch FAQ: What is OpenSearch?

← Back to Blog