Apache Spark: A Unified Engine for Big Data Processing

Apache Spark started as a project to design a unified engine for distributed data processing. It uses RDDs (Resilient Distributed Datasets) to extend the MapReduce model and use it as it's own. Spark can capture a wide range of processing workloads: SQL, streaming, machine learning and graph processing with similar performance than using it with a specialized engine, but treating them as libraries.

There are several benefits that one gets while using Spark. One of them is the ease of use when developing applications, apart from that it is more efficient to combine processing tasks, and finally it provides the ability to use new applications (such as interactive queries on a graph and streaming machine learning).

Programming Model

RDDs are fault-tolerant collections of objects partitioned across a cluster that can be used in parallel. They are created by applying transformations such as map, filter and groupBy to their data. Spark uses functional programming for it's Scala, Java, Python, and R APIs. Also, Spark evaluates RDDs lazily. RDDs also provide explicit support for data sharing among computations.

Higher-Level Libraries

Libraries

With SparkSQL one can implement sql queries on spark, using techniques similar to analytical databases. These support columnar storage, cost-based optimization, and code generation for query execution. You can process streaming data using "discretized streams", which inputs data into small batches that regularly get combined with state stored in RDDs. Another higher-level api that you can use is GraphX for computing graphs, it is similar to Pregel and GraphLab. With MLlib, one can work with more than 50 common algorithms for distributed model training for machine learning.

Benefits

All of Spark's libraries operate on RDDs as the data abstraction, which makes them easier to combine in applications. Also, because these apps run on the same engine, they should theoretically lose performance, but with the implementation of different optimizations said before, the performance drop is almost null. Spark has the ability to use batch processing and it gets used in large datasets, including Extract-Transform-Load workloads to convert data from a raw format to structured one used for offline machine learning models in training. Apart from these, spark implements interactive queries, which helps the user with relational queries, while interfaces such as the one for Scala or Python get used to interact with queries over a visual environment. Spark provides real-time processing for both analytics and real time decision-making applications. Lastly, Spark is also used in scientific domains such as large-scale spam detection, image processing, and genomic data processing.