What is Apache Hadoop?

Apache Hadoop is an open-source software framework developed by Douglas Cutting, then at Yahoo, that provides the highly reliable distributed processing of large data sets using simple programming models. Hadoop overcame the scalability limitations of Nutch, and is built on clusters of commodity computers, providing a cost-effective solution for storing and processing massive amounts of structured, semi-structured and unstructured data with no format requirements.

A data lake architecture including Hadoop can offer a flexible data management solution for your big data analytics initiatives. Because Hadoop is an open-source project and follows a distributed computing model, it can offer budget-saving pricing for a big data software and storage solution.

Hadoop can also be installed on cloud servers to better manage the compute and storage resources required for big data. For greater convenience, the Linux OS agent, UNIX OS agent, and Windows OS agent are pre-configured and can be started automatically. Leading cloud vendors such as Amazon Web Services (AWS) and Microsoft Azure offer solutions. Cloudera supports Hadoop workloads both on-premises and in the cloud, including options for one or more public cloud environments from multiple vendors. Use Hadoop monitoring APIs to add, update, delete and view the clusters and services on the clusters, and for all other types of monitoring on Hadoop.

The Hadoop ecosystem

The Hadoop framework, built by the Apache Software Foundation, includes:

- Hadoop Common: The common utilities and libraries that support the other Hadoop modules. Also known as Hadoop Core.

- Hadoop HDFS (Hadoop Distributed File System): A distributed file system for storing application data on commodity hardware. HDFS was designed to provide fault tolerance for Hadoop and it provides high aggregate data bandwidth and high-throughput access to data. By default, data blocks are replicated across multiple nodes at load or write time. The degree of replication is configurable: the default replication is three. The HDFS architecture features a NameNode to manage the file system namespace and file access and multiple DataNodes to manage data storage. By enabling high availability, a secondary node can be used when an active node goes down.

- Hadoop YARN: Open-source Apache Hadoop YARN is a framework for job scheduling and cluster resource management that can be used with IBM® Spectrum Symphony on Linux® and Linux on POWER®. YARN stands for Yet Another Resource Negotiator. It supports more workloads, such as interactive SQL, advanced modeling and real-time streaming.

- Hadoop MapReduce: A YARN-based system that stores data on multiple sources and powers for parallel processing of large amounts of data. Multiple optimization techniques are available for MapReduce to speed jobs.

- Hadoop Ozone: A scalable, redundant and distributed object store designed for big data applications.

Supporting Apache projects

Enhance Hadoop with additional open-source software projects.

- Ambari
  A web-based tool for provisioning, managing and monitoring Hadoop clusters.

- Avro
  A data serialization system.

- Cassandra
  A scalable, NoSQL database designed to have no single point of failure.

- Chukwa
  A data collection system for monitoring large distributed systems; built on top of HDFS and MapReduce.

- Flume
  A service for collecting, aggregating and moving large amounts of streaming data into HDFS.

- HBase
  A scalable, non-relational distributed database that supports structured data storage for very large tables.

- Hive
  A data warehouse infrastructure for data querying, metadata storage for tables and analysis in a SQL-like interface.

- Mahout
  A scalable machine learning and data mining library.

- Oozie
  A Java-based workload scheduler to manage Hadoop jobs.

- Pig

A high-level data flow language and execution framework for parallel computation.

- Sqoop
  A tool for efficiently transferring data between Hadoop and structured data stores such as relational databases.

- Submarine
  A unified AI platform for running machine learning and deep learning workloads in a distributed cluster.

- Tez
  A generalized data flow programming framework, built on YARN; being adopted within the Hadoop ecosystem to replace MapReduce.

- ZooKeeper
  A high performance coordination service for distributed applications.

Hadoop for developers

Apache Hadoop was written in Java, but depending on the big data project, developers can program in their choice of language, such as Python, R or Scala. The included Hadoop Streaming utility enables developers to create and execute MapReduce jobs with any script or executable as the mapper or the reducer.

Spark vs. Hadoop

Apache Spark is often compared to Hadoop as it is also an open-source framework for big data processing. In fact, Spark was initially built to improve the processing performance and extend the types of computations possible with Hadoop MapReduce. Spark uses in-memory processing, which means it is vastly faster than the read/write capabilities of MapReduce.

While Hadoop is best for batch processing of huge volumes of data, Spark supports both batch and real-time data processing and is ideal for streaming data and graph computations. Both Hadoop and Spark have machine learning libraries, but again, because of the in-memory processing, Spark’s machine learning is much faster.

Hadoop use cases

Better data-driven decisions: Integrate real-time data (streaming audio, video, social media sentiment and clickstream data) and other semi-structured and unstructured data not used in a data warehouse or relational database. More comprehensive data provides more accurate decisions.

Improved data access and analysis: Drive real-time, self-service access for your data scientist, line of business (LOB) owners and developers. Hadoop can fuel data science, an interdisciplinary field that uses data, algorithms, machine learning and AI for advanced analysis to reveal patterns and build predictions.

Data offload and consolidation: Streamline costs in your enterprise data centers by moving "cold" data not currently in use to a Hadoop-based distribution for storage. Or consolidate data across the organization to increase accessibility and decrease costs.