HADOOP ECOSYSTEM AND COMPONENTS



Hadoop ecosystem overview

Remember that Hadoop is a framework. If Hadoop was a house, it wouldn't be a very comfortable place to live. It would provide walls, windows, doors, pipes, and wires. The Hadoop ecosystem provides the furnishings that turn the framework into a comfortable home for big data activity that reflects your specific needs and tastes.

The Hadoop ecosystem includes both official Apache open source projects and a wide range of commercial tools and solutions. Some of the best-known open source examples include Spark, Hive, Pig, Oozie and Sqoop. Commercial Hadoop offerings are even more diverse and include platforms and packaged distributions from vendors such as Cloudera, Hortonworks, and MapR, plus a variety of tools for specific Hadoop development, production, and maintenance tasks.



Most of the solutions available in the Hadoop ecosystem are intended to supplement one or two of

Hadoop's four core elements (HDFS, MapReduce, YARN, and Common). However, the commercially available framework solutions provide more comprehensive functionality. The sections below provide a closer look at some of the more prominent components of the Hadoop ecosystem, starting with the Apache projects

(This article is part of our Hadoop Guide. Use the right-hand menu to navigate.)

Apache open source Hadoop ecosystem elements

The Apache Hadoop project actively supports multiple projects intended to extend Hadoop's capabilities and make it easier to use. There are several top-level projects to create development tools as well as for managing Hadoop data flow and processing. Many commercial third-party solutions build on the technologies developed within the Apache Hadoop ecosystem.

Spark, Pig, and Hive are three of the best-known Apache Hadoop projects. Each is used to create applications to process Hadoop data. While there are a lot of articles and discussions about whether Spark, Hive or Pig is better, in practice many organizations do not only use a single one because each is optimized for specific functions.

Spark

Spark is both a programming model and a computing model. It provides a gateway to in-memory computing for Hadoop, which is a big reason for its popularity and wide adoption. Spark provides an alternative to MapReduce that enables workloads to execute in memory, instead of on disk. Spark accesses data from HDFS but bypasses the MapReduce processing framework, and thus eliminates the resource-intensive disk operations that MapReduce requires. By using in-memory computing, Spark workloads typically run between 10 and 100 times faster compared to disk execution.

Spark can be used independently of Hadoop. However, it is used most commonly with Hadoop as an alternative to MapReduce for data processing. Spark can easily coexist with MapReduce and with other ecosystem components that perform other tasks.

Spark is also popular because it supports SQL, which helps overcome a shortcoming in core Hadoop technology. The Spark programming environment works interactively with Scala, Python, and R shells. It has been used for data extract/transform/load (ETL) operations, stream processing, machine learning development and with the Apache GraphX API for graph computation and display. Spark can run on a variety of Hadoop and non-Hadoop clusters, including Amazon S3.

Hive

Hive is data warehousing software that addresses how data is structured and queried in distributed Hadoop clusters. Hive is also a popular development environment that is used to write queries for data in the Hadoop environment. It provides tools for ETL operations and brings some SQL-like capabilities to the environment. Hive is a declarative language that is used to develop applications for the Hadoop environment, however it does not support real-time queries.

Hive has several components, including:

- **HCatalog** – Helps data processing tools read and write data on the grid. It supports MapReduce and Pig.

- **WebHCat** – Lets you use an HTTP/REST interface to run MapReduce, Yarn, Pig, and Hive jobs.
- **HiveQL** – Hive's query language intended as a way for SQL developers to easily work in Hadoop. It is similar to SQL and helps both structure and query data in distributed Hadoop clusters.

Hive queries can run from the Hive shell, JDBC, or ODBC. MapReduce (or an alternative) breaks down HiveQL statements for execution across the cluster.

Hive also allows MapReduce-compatible mapping and reduction software to perform more sophisticated functions. However, Hive does not allow row-level updates or support for real-time queries, and it is not intended for OLTP workloads. Many consider Hive to be much more effective for processing structured data than unstructured data, for which Pig is considered advantageous.

Pig

Pig is a procedural language for developing parallel processing applications for large data sets in the Hadoop environment. Pig is an alternative to Java programming for MapReduce, and automatically generates MapReduce functions. Pig includes Pig Latin, which is a scripting language. Pig translates Pig Latin scripts into MapReduce, which can then run on YARN and process data in the HDFS cluster. Pig is popular because it automates some of the complexity in MapReduce development.

Pig is commonly used for complex use cases that require multiple data operations. It is more of a processing language than a query language. Pig helps develop applications that aggregate and sort data and supports multiple inputs and exports. It is highly customizable, because users can write their own functions using their preferred scripting language. Ruby, Python and even Java are all supported. Thus, Pig has been a popular option for developers that are familiar with those languages but not with MapReduce. However, SQL developers may find Hive easier to learn.

HBase

HBase is a scalable, distributed, NoSQL database that sits atop the HFDS. It was designed to store structured data in tables that could have billions of rows and millions of columns. It has been deployed to power historical searches through large data sets, especially when the desired data is contained within a large amount of unimportant or irrelevant data (also known as sparse data sets). It is also an underlying technology behind several large messaging applications, including Facebook's.

HBase is not a relational database and wasn't designed to support transactional and other real-time applications. It is accessible through a Java API and has ODBC and JDBC drivers. HBase does not support SQL queries, however there are several SQL support tools available from the Apache project and from software vendors. For example, Hive can be used to run SQL-like queries in HBase.

Oozie

Oozie is the workflow scheduler that was developed as part of the Apache Hadoop project. It manages how workflows start and execute, and also controls the execution path. Oozie is a server-based Java web application that uses workflow definitions written in hPDL, which is an XML Process Definition Language similar to JBOSS JBPM jPDL. Oozie only supports specific workflow types, so other workload schedulers are commonly used instead of or in addition to Oozie in Hadoop

environments.

Sqoop

Think of Sqoop as a front-end loader for big data. Sqoop is a command-line interface that facilitates moving bulk data from Hadoop into relational databases and other structured data stores. Using Sqoop replaces the need to develop scripts to export and import data. One common use case is to move data from an enterprise data warehouse to a Hadoop cluster for ETL processing. Performing ETL on the commodity Hadoop cluster is resource efficient, while Sqoop provides a practical transfer method.

Other Apache Hadoop-related open source projects

Here is how the Apache organization describes some of the other components in its Hadoop ecosystem.

- **Ambari** – A web-based tool for provisioning, managing, and monitoring Apache Hadoop clusters which includes support for Hadoop HDFS, Hadoop MapReduce, Hive, HCatalog, HBase, ZooKeeper, Oozie, Pig, and Sqoop.
- **Avro** – A data serialization system.
- **Cassandra** – A scalable multi-master database with no single points of failure.
- **Chukwa** – A data collection system for managing large distributed systems.
- **Impala** – The open source, native analytic database for Apache Hadoop. Impala is shipped by Cloudera, MapR, Oracle, and Amazon.
- **Flume** – A distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of streaming event data.
- **Kafka** – A messaging broker that is often used in place of traditional brokers in the Hadoop environment because it is designed for higher throughput and provides replication and greater fault tolerance.
- **Mahout** – A scalable machine learning and data mining library.
- **Tajo** – A robust big data relational and distributed data warehouse system for Apache Hadoop. Tajo is designed for low-latency and scalable ad-hoc queries, online aggregation, and ETL on large-data sets stored on HDFS and other data sources. By supporting SQL standards and leveraging advanced database techniques, Tajo allows direct control of distributed execution and data flow across a variety of query evaluation strategies and optimization opportunities.
- **Tez** – A generalized data-flow programming framework, built on Hadoop YARN, which provides a powerful and flexible engine to execute an arbitrary DAG of tasks to process data for both batch and interactive use-cases. Tez is being adopted by Hive, Pig and other frameworks in the Hadoop ecosystem, and also by other commercial software (e.g. ETL tools), to replace MapReduce as the underlying execution engine.
- **Zookeper** – A high-performance coordination service for distributed applications.

The ecosystem elements described above are all open source Apache Hadoop projects. There are numerous commercial solutions that use or support the open source Hadoop projects. Some of the more prominent ones are described in the following sections.

Commercial Hadoop distributions

Hadoop can be downloaded from www.hadoop.apache.org and used for free, which thousands of organizations have done. There are also commercial distributions that combine core Hadoop technology with additional features, functionality and documentation. The leading commercial distribution Hadoop vendors include Cloudera, Hortonworks, and MapR. There are also many more less comprehensive, more task-specific tools for the Hadoop environment, such as developer tools and job schedulers.