HADOOP TUTORIAL FOR BEGINNERS: HADOOP BASICS

What is Hadoop?

Hadoop (the full proper name is Apache™ Hadoop®) is an open-source framework that was created to make it easier to work with big data. It provides a method to access data that is distributed among multiple clustered computers, process the data, and manage resources across the computing and network resources that are involved. “Hadoop” commonly refers to the core technology that consists of the four main components described below, but is also frequently used in reference to the entire ecosystem of supporting technologies and applications.

“Hadoop” also is often used interchangeably with “big data,” but it shouldn’t be. Hadoop is a framework for working with big data. It is part of the big data ecosystem, which consists of much more than Hadoop itself.

Hadoop is a distributed framework that makes it easier to process large data sets that reside in clusters of computers. Because it is a framework, Hadoop is not a single technology or product. Instead, Hadoop is made up of four core modules that are supported by a large ecosystem of supporting technologies and products. The modules are:

- **Hadoop Distributed File System (HDFS™) –**Provides access to application data. Hadoop can also work with other file systems, including FTP, Amazon S3 and Windows Azure Storage Blobs (WASB), among others.
- **Hadoop YARN –** Provides the framework to schedule jobs and manage resources across the cluster that holds the data
- **Hadoop MapReduce –** A YARN-based parallel processing system for large data sets.
- **Hadoop Common –** A set of utilities that supports the three other core modules.

Some of the well-known Hadoop ecosystem components include Oozie, Spark, Sqoop, Hive and Pig.

What Hadoop isn't

In this tutorial for beginners, it's helpful to understand what Hadoop is by knowing what it is not.

- Hadoop is not "big data" – the terms are sometimes used interchangeably, but they shouldn't be. Hadoop is a framework for processing big data.
- Hadoop is not an operating system (OS) or packaged software application.
- Hadoop is not a brand name. It is an open source project, although "Hadoop" may be used as part of registered brand names.

What's with the name?

Hadoop was originally developed by Doug Cutting and Mike Cafarella. According to lore, Cutting named the software after his son's toy elephant. An image of an elephant remains the symbol for Hadoop.

Core elements of Hadoop

There are four basic elements to Hadoop: HDFS; MapReduce; YARN; Common.

HDFS

Hadoop works across clusters of commodity servers. Therefore there needs to be a way to coordinate activity across the hardware. Hadoop can work with any distributed file system, however the Hadoop Distributed File System is the primary means for doing so and is the heart of Hadoop technology. HDFS manages how data files are divided and stored across the cluster. Data is divided into blocks, and each server in the cluster contains data from different blocks. There is also some built-in redundancy.

YARN

It would be nice if **YARN** could be thought of as the string that holds everything together, but in an environment where terms like Oozie, tuple and Sqoop are common, of course it's not that simple.

YARN is an acronym for Yet Another Resource Negotiator. As the full name implies, YARN helps manage resources across the cluster environment. It breaks up resource management, job scheduling, and job management tasks into separate daemons. Key elements include the **ResourceManager (RM)**, the **NodeManager (NM)** and the ApplicationMaster (AM).

Think of the ResourceManager as the final authority for assigning resources for all the applications in the system. The NodeManagers are agents that manage resources (e.g. CPU, memory, network, etc.) on each machine. NodeManagers report to the ResourceManager. ApplicationMaster serves as a library that sits between the two. It negotiates resources with ResourceManager and works with one or more NodeManagers to execute tasks for which resources were allocated.

MapReduce

MapReduce provides a method for parallel processing on distributed servers. Before processing data, MapReduce converts that large blocks into smaller data sets called **tuples**. Tuples, in turn, can be organized and processed according to their key-value pairs. When MapReduce processing is complete, HDFS takes over and manages storage and distribution for the output. The shorthand version of MapReduce is that it breaks big data blocks into smaller chunks that are easier to work with.

The "Map" in MapReduce refers to the **Map Tasks** function. Map Tasks is the process of formatting data into key-value pairs and assigning them to nodes for the "Reduce" function, which is executed by **Reduce Tasks**, where data is reduced to tuples. Both Map Tasks and Reduce Tasks use **worker nodes** to carry out their functions.

JobTracker is a component of the MapReduce engine that manages how client applications submit MapReduce jobs. It distributes work to **TaskTracker** nodes. TaskTracker attempts to assign processing as close to where the data resides as possible.

Note that MapReduce is not the only way to manage parallel processing in the Hadoop environment.

Common

Common, which is also known as **Hadoop Core**, is a set of utilities that support the other Hadoop components. Common is intended to give the Hadoop framework ways to manage typical (common) hardware failures.

What is 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.

How is Hadoop related to big data?

Big data is becoming a catchall phrase, while Hadoop refers to a specific technology framework. Hadoop is a gateway that makes it possible to work with big data, or more specifically, large data sets that reside in a distributed environment. One way to define big data is data that is too big to be processed by relational database management systems (RDBMS). Hadoop helps overcome RDBMS limitations, so big data can be processed.

Perhaps a more important question than How is Hadoop related to big data? is How does Hadoop relate to other big data technologies? The relationships between the core Hadoop modules and the technologies and solutions that complement and compete with them are covered in more depth in the Hadoop Ecosystem section of this guide.

What does Hadoop replace?

In many cases Hadoop hasn't replaced anything, because the things it is being used for simply were not done before (especially with unstructured data) because of computing system limitations. Relational databases and distributed file systems each do parts of what Hadoop can do, but operate on a much smaller scale. Again, a more instructive question is *Which elements of Hadoop can be replaced or enhanced by other technologies and products in the ecosystem?*

See how CARFAX uses Big Data and Hadoop

Who uses Hadoop? Why?

Any list of how Hadoop is being used and the organizations that are using it would become out of date in the time it takes to save the file. The Apache Hadoop organization maintains a list of Hadoop users that is extensive, but not comprehensive, at http://wiki.apache.org/hadoop/PoweredBy. Some of the prominent users listed include Amazon, EBay, Facebook, Google, IBM, LinkedIn, the New York Times, Rackspace and Yahoo.

To generalize, Hadoop has found a home in industries and organizations characterized by having large data sets, time-sensitive data, and data that could provide insight to improve performance or revenue. More specifically, the financial services, telecommunications, utilities/energy, and retail industries have been early Hadoop adopters and innovators, along with some government and other public sector organizations.

Hadoop examples: 5 real-world use cases

Hadoop Examples & Use Cases >