Chapter 3
Hadoop History and Architecture

ABSTRACT

As the name indicates, this chapter explains the evolution of Hadoop. Doug Cutting started a text search library called Lucene. After joining Apache Software Foundation, he modified it into a web crawler called Apache Nutch. Then Google File System was taken as reference and modified as Nutch Distributed File System. Then Google's MapReduce features were also integrated and Hadoop was framed. The whole path from Lucene to Apache Hadoop is illustrated in this chapter. Also, the different versions of Hadoop are explained. The procedure to download the software is explained. The mechanism to verify the downloaded software is shown. Then the architecture of Hadoop is detailed. The Hadoop cluster is a set of commodity machines grouped together. The arrangement of Hadoop machines in different racks is shown. After reading this chapter, the reader will understand how Hadoop has evolved and its entire architecture.

INTRODUCTION

Hadoop is an open source framework used for storing and processing big data. It is developed by Apache Software Foundation. Hadoop environment can be setup with commodity hardware alone. It is a framework that supports distributed environment with cluster of commodity machines. It can work with single server or can scale up including thousands of commodity machines.

DOI: 10.4018/978-1-5225-3790-8.ch003

Hadoop has undergone number of revisions also. This chapter gives the novice users an idea about how Hadoop was initiated and what are the major revisions of it. Also this chapter describes in detail the architecture of Hadoop.

BACKGROUND

The most acute information management challenges stem from organizations (e.g., enterprises, government agencies, libraries, "smart" homes) relying on a large number of diverse, interrelated data sources, but having no way to manage their *dataspaces* (Franklin, Halevy, & Maier, 2005) in a convenient, integrated, or principled fashion. Michael Franklin et.al, (2005) highlighted the need for storage systems to accept all data formats and to provide APIs for data access that evolve based on the storage system's understanding of the data.

In the past years, (Dean & Ghemawat, 2004) at Google have implemented hundreds of special- purpose computations that process large amounts of raw data, such as crawled documents, web request logs, etc., to compute various kinds of derived data, such as inverted indices, various representations of the graph structure of web documents, summaries of the number of pages crawled per host, the set of most frequent queries in a given day, etc. Most such computations are conceptually straightforward. However, the input data is usually large and the computations have to be distributed across hundreds or thousands of machines in order to finish in a reasonable amount of time.

Robert Kallman et.al, (2008) developed H-Store, a next-generation OLTP system that operates on a distributed cluster of shared-nothing machines where the data resides entirely in main memory. But it needs a separate database design for the attributes- Table replication and Data partitioning. To solve the above problems, (Chang, Dean, Ghemawat, Hsieh, Wallach, Burrows… Gruber, 2008) developed BigTable which is distributed storage system for maintaining structured data of petabytes size across thousands of commodity servers. Later an open source equivalent to BigTable was created and it was called "Hadoop". Hadoop is an open source platform that brings the ability to cheaply process large amounts of data and it is more suitable for storing voluminous unstructured data.