Volume 4, Issue 1, January 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Aspects and Infrastructure of Big Data

Umasri.M.L (PG), *Sri Ramakrishna Engg Clg, India*

Shyamalagowri.D (LEC) *Ranganathan Engg College India*

Suresh Kumar.S (AP) *Sri Ramakrishna Engg Clg, India*

Abstract: Big Data is a new term used to identify the datasets that due to their large size and complexity, we cannot manage them with current methodologies. Every day we create 2.5 quintillion byte of data. This data exponential growth in the volume and detail of data captured by enterprises, the rise of multimedia, social media and Online Social Network (OSN). In this paper we see details about aspects of Big Data and how to reduce the data by Hadoop methodology and then what are tool available.

Keywords:

1. INTRODUCTION

What is big data? Big data is data that exceeds the processing capacity of conventional database systems. The data is too big, moves too fast, or doesn't fit the structures of your database architectures. The hot IT buzzword of 2012, big data has become viable as cost effective approaches have emerged to tame the volume, velocity, and variability of massive data. Input data to big data systems could be chatter from social networks, web server logs, traffic flow sensors, satellite imagery, broadcast audio streams, banking transactions, MP3s of rock music, the content of web pages, scans of government documents, GPS trails, telemetry from automobiles, financial market data, the list goes on. Are these all really the same thing? To clarify matters, the three Vs of volume, velocity, and variety are commonly used to characterize different aspects of big data [1].

Table-1 Different aspects of Big Data.

Volume	There is more data than ever before; its size continues increasing, but not the percent of data that our tools can process.
Variety	There are many different types of data, as text, sensor data, audio, video, graph, and more.
Velocity	Data is arriving continuously as streams of data, and we are interested in obtaining useful information from it in real time.
Variability	There are changes in the structure of the data and how users want to interpret that data.
Value	Business value that gives organization a compelling advantage, due to the ability of making decisions based in answering questions that were previously considered beyond reach.

2. Big Data infrastructure

Big Data infrastructure deals with Hadoop.

2.1 What Is Apache Hadoop?

Apache Hadoop has been the driving force behind the growth of the big data industry. Hadoop brings the ability to cheaply process large amounts of data, regardless of its structure. By large, we mean from 10-100 gigabytes and above. Existing enterprise data warehouses and relational databases excel at processing structured data and can store massive amounts of data, though at a cost: This requirement for structure restricts the kinds of data that can be processed, and it imposes an inertia that makes data warehouses unsuited for agile exploration of massive heterogeneous data. The amount of effort required to warehouse data often means that valuable data sources in organizations are never mind. This is where Hadoop can make a big difference.

2.2 The Core of Hadoop: MapReduce

Created at Google in response to the problem of creating web search indexes, the Map Reduce framework is the powerhouse behind most of today's big data processing. In addition to Hadoop, you'll find Map-Reduce inside MPP and NoSQL databases, such as Vertica or MongoDB.

The important innovation of Map Reduce is the ability to take a query over a dataset, divide it, and run it in parallel over multiple nodes. Distributing the computation solves the issue of data too large to fit onto a single machine. Combine this technique with commodity Linux servers and you have a cost-effective alternative to massive computing arrays. At its core, Hadoop is an open source MapReduce implementation. Funded by Yahoo, it emerged in 2006 and, according to its creator Doug Cutting, reached "web scale" capability in early 2008.

2.3 Hadoop's Lower Levels: HDFS and MapReduce

The ability of Map Reduce to distribute computation over multiple servers. For that computation to take place, each server must have access to the data. This is the role of HDFS, the Hadoop Distributed File System. HDFS and Map Reduce are robust. Servers in a Hadoop cluster can fail and not abort the computation process. HDFS ensures data is replicated with redundancy across the cluster. On completion of a calculation, a node will write its results back into HDFS. There are no restrictions on the data that HDFS stores. Data may be unstructured and schema less. By contrast, relational databases require that data be structured and schemas be defined before storing the data. With HDFS, making sense of the data is the responsibility of the developer's code. Programming Hadoop at the MapReduce level is a case of working with the Java APIs, and manually loading data files into HDFS.

2.4 Improving Programmability: Pig and Hive

Working directly with Java APIs can be tedious and error prone. It also restricts usage of Hadoop to Java programmers. Hadoop offers two solutions for making Hadoop programming easier.

- Pig is a programming language that simplifies the common tasks of working with Hadoop: loading data, expressing transformations on the data, and storing the final results. Pig's built-in operation scan makes sense of semi-structured data, such as log files, and the language is extensible using Java to add support for custom data types and transformations.
- Hive enables Hadoop to operate as a data warehouse. It superimpose structure on data in HDFS and then permits queries over the data using a familiar SQL-like syntax.

Choosing between Hive and Pig can be confusing. Hive is more suitable for data warehousing tasks, with predominantly static structure and the need for frequent analysis. Hive's closeness to SQL makes it an ideal point of integration between Hadoop and other business intelligence tools.

Pig gives the developer more agility for the exploration of large datasets, allowing the development of succinct scripts for transforming data flows for incorporation into larger applications. Pig is a thinner layer over Hadoop than Hive, and its main advantage is to drastically cut the amount of code needed compared to direct use of Hadoop's Java APIs. As such, Pig's intended audience remains primarily the software developer.

2.5 Improving Data Access: HBase, Sqoop, and Flume

At its heart, Hadoop is a batch-oriented system. Data are loaded into HDFS, processed, and then retrieved. This is somewhat of a computing throwback, and often, interactive and random access to data is required. Enter HBase, a column-oriented database that runs on top of HDFS. Modeled after Google's BigTable, the project's goal is to host billions of rows of data for rapid access. MapReduce can use HBase as both a source and a destination for its computations, and Hive and Pig can be used in combination with HBase.

In order to grant random access to the data, HBase does impose a few restrictions: Hive performance with HBase is 4-5 times slower than with plain HDFS, and the maximum amount of data you can store in HBase is approximately a petabyte, versus HDFSlimit of over 30PB. HBase is ill-suited to ad-hoc analytics and more appropriate for integrating big data as part of a larger application. Use cases include logging, counting, and storing time-series data.

Improved interoperability with the rest of the data world is provided by Sqoop and Flume. Sqoop is a tool designed to import data from relational databases into Hadoop, either directly into HDFS or into Hive. Flume is designed to import streaming flows of log data directly into HDFS. Hive's SQL friendliness means that it can be used as a point of integration with the vast universe of database tools capable of making connections via JBDC or ODBC database drivers.

2.6 Coordination and Workflow: Zookeeper and Oozie

*W*ith a growing family of services running as part of a Hadoop cluster, there's a need for coordination and naming services. As computing nodes can come and go, members of the cluster need to synchronize with each other, know where to access services, and know how they should be configured. This is the purpose of Zookeeper.

Production systems utilizing Hadoop can often contain complex pipelines of transformations, each with dependencies on each other. For example, the arrival of a new batch of data will trigger an import, which must then trigger recalculations in dependent datasets. The Oozie component provides features to manage the workflow and dependencies, removing the need for developers to code custom solutions.

2.7 Management and Deployment: Ambari and Whirr

One of the commonly added features incorporated into Hadoop by distributors such as IBM and Microsoft is monitoring and administration. Though in an early stage, Ambari aims to add these features to the core Hadoop project. Ambari is intended to help system administrators deploy and configure Hadoop, upgrade clusters, and monitor services. Through an API, it may be integrated with other system management tools. Though not strictly part of Hadoop, Whirr is a highly complementary component. It offers a way of running services, including Hadoop, on cloud platforms. Whirr is cloud neutral and currently supports the Amazon EC2 and Rackspace services.

2.8 Machine Learning: Mahout

Every organization's data are diverse and particular to their needs. However, there is much less diversity in the kinds of analyses performed on that data. The Mahout project is a library of Hadoop implementations of common analytical computations. Use cases include user collaborative filtering, user recommendations, clustering, and classification.

Table-2 The Hadoop Bestiary

Ambari	Deployment, configuration and monitoring
Flume	Collection and import of log and event data
HBase	Column-oriented database scaling to billions of rows
HCatalog	Schema and data type sharing over Pig, Hive and MapReduce
HDFS	Distributed redundant file system for Hadoop
Hive	Data warehouse with SQL-like access
Mahout	Library of machine learning and data mining algorithms
Map Reduce	Parallel computation on server clusters
Pig	High-level programming language for Hadoop computations
Oozie	Orchestration and workflow management
Sqoop	Imports data from relational databases
Whirr	Cloud-agnostic deployment of clusters
Zookeeper	Configuration management and coordination

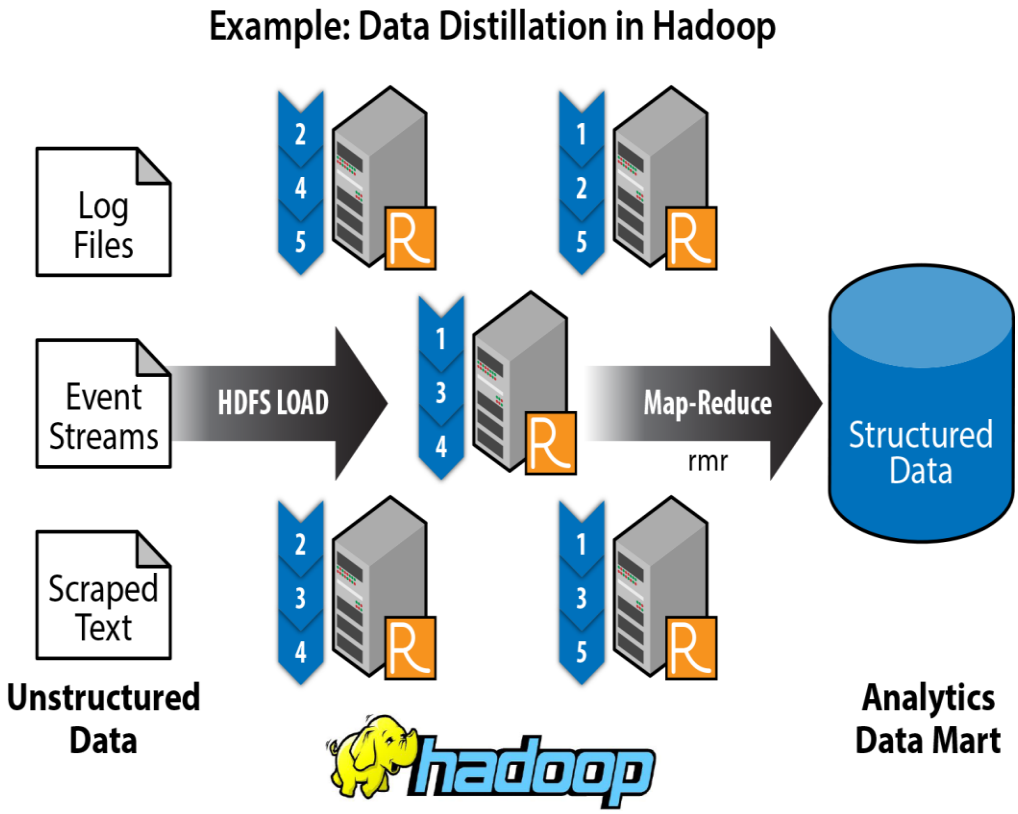

Figure 1 From David Smith's presentation, "Real-Time Big Data Analytics: From Deployment to Production"

3. Tools: Open Source Revolution

The Big Data phenomenon is intrinsically related to the open source software revolution. Large companies as Face book, Yahoo!, Twitter, and LinkedIn benefit and contribute working on open source projects. Big Data infrastructure deals with Hadoop, and other related software as:

3.1 Apache Hadoop [2]*:* software for data-intensive distributed applications, based in the Map Reduce programming model and a distributed file system called Hadoop Distributed File system (HDFS). Hadoop allows writing applications that rapidly process large amounts of data in parallel on large clusters of compute nodes.

A Map Reduce job divides the input dataset into independent subsets that are processed by map tasks in parallel. This step of mapping is then followed by a step of reducing tasks. These reduce tasks use the output of the maps to obtain the final result of the job.

3.2 Apache Hadoop related projects [3]: Apache Pig, Apache Hive, Apache HBase, Apache ZooKeeper, Apache Cassandra, Cascading, Scribe and many others.

3.3 Apache S4 [4]: platform for processing continuous data streams. S4 is designed specifically for managing data streams. S4 apps are designed combining streams and processing elements in real time.

3.4 Storm [5]: software for streaming data-intensive distributed applications, similar to S4, and developed by Nathan Marz at Twitter.

In Big Data Mining, there are many open source initiatives. The most popular are the following:

3.5 Apache Mahout [6]: Scalable machine learning and data mining open source software based mainly in Hadoop. It has implementations of a wide range of machine learning and data mining algorithms: clustering, classification, collaborative filtering and frequent pattern mining.

3.6 R [7]: open source programming language and software environment designed for statistical computing and visualization. R was designed by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand beginning in 1993 and is used for statistical analysis of very large data sets.

3.7 MOA [8]: Stream data mining open source software to perform data mining in real time. It has implementations of classification, regression; clustering and frequent item set mining and frequent graph mining. It started as a project of the Machine Learning group of University of Waikato, New Zealand, famous for the WEKA software. The streams framework provides an environment for defining and running stream processes using simple XML based definitions and is able to use MOA, Android and Storm. SAMOA [1] is a new upcoming software project for distributed stream mining that will combine S4 and Storm with MOA.

3.8 VowpalWabbit [9]: open source project started at Yahoo! Research and continuing at Microsoft Research to design a fast, scalable, useful learning algorithm. VW is able to learn from terafeature datasets. It can exceed the throughput of any single machine network interface when doing linear learning, via parallel learning.

4. Conclusions

Big Data is going to continue growing during the next years, and each data scientist will have to manage much more amount of data every year. This data is going to be more diverse, larger, and faster. Big Data is becoming the new Final Frontier for scientific data research and for business applications. We are at the beginning of a new era where Big Data mining will help us to discover knowledge that no one has discovered before. Everybody is warmly invited to participate in this intrepid journey.

Reference

[1] Gartner, http://www.gartner.com/it-glossary/bigdata.
[2] Apache Hadoop, http://hadoop.apache.org.
[3] P. Zikopoulos, C. Eaton, D. deRoos, T. Deutsch, and G. Lapis. IBM Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data. McGraw-Hill Companies, Incorporated, 2011.
[4] L. Neumeyer, B. Robbins, A. Nair, and A. Kesari.S4: Distributed Stream Computing Platform. In ICDM Workshops, pages 170-177.
[5] Storm, http://storm-project.net.
[6] Apache Mahout, http://mahout.apache.org.
[7] R Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.
[8] A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer. MOA: Massive Online Analysis http://moa.cms.waikato.ac.nz/. Journal of Machine Learning Research (JMLR), 2010.
[9] J. Langford. VowpalWabbit, http://hunch.net/~vw/, 2011.

