RESEARCH ARTICLE OPEN ACCESS

Semantic Web Repository Maintenance on Hadoop

Dr. K. Sridevi
Assistant Professor
Department of Computer Science
Nehru Memorial College
Puthanampatti, Trichy District
Tamilnadu – India

ABSTRACT
The rapid growth of data on the web demands new strategies for processing and analysing information. Such large volume of un-structured (or semi structured) and structured data that gets created from various applications such as emails, web logs, social media is known as "Big Data". This kind of data exceeds the processing capacity of conventional database systems. The keyword based information retrieval technology of these data fails to integrate information spread over different resources. This technology does not use the semantics, to overcome this problem in Web, the next-generation Web, which Tim Berners-Lee and others call the "Semantic Web," aims at allowing machines to process information automatically and gives focus on semantics of the content. Resource Description Format (RDF) plays an important role on acting as Semantic Web Repository such as RDF graph and used to maintain Big Data on Semantic Web. Since the amount of RDF data is increasing, managing the same becomes a matter of utmost importance. Managing the RDF data has encountered the problem of scalability. MapReduce has been introduced as a standard framework to process a high amount of data in parallel and to some extent it has solved the scalability problem. This paper provides the basics of Big Data, the most widely used Big Data Analytics technique, called Hadoop's MapReduce to manage Big Data.

Kewords:- Semantic Web, RDF, Big Data, Hadoop, Map Reduce.

I. INTRODUCTION

Conventional direct keyword based information retrieval mechanism cannot allow machines to process information automatically and gives focus on semantics of the content. Resource description framework (RDF) on Semantic Web offers an efficient way to reduce the amount of information overload by encoding the structure of a specific domain and offering easier and meaningful access to the information for the users. RDF is a kind of data model which is used to represent the information about WWW resources, and to make semantic data and the link between them [1]. RDF can be considered as a directed labeled graph. In recent years, RDF has highly been regarded in the industry and university. In RDF model, every resource is described as a triple (subject, predicate, object). This model is schema-relaxed, i.e. it provides the possibility of mixing two data with different fundamental structures. The growth of RDF repository becomes big as the content on web increases, termed as Big Data [2]. Hence, going behind the technology of Big Data is essential.

Big data technologies are important in providing more accurate analysis, which may lead to more concrete decision-making resulting in greater operational efficiencies, cost reductions, and reduced risks. The technologies that handle big data can be classified into the two classes such as operational big data and analytical big data [3].

Operational big data includes systems like MongoDB that provide operational capabilities for real-time, interactive workloads where data is primarily captured and stored. NoSQL big data systems are designed to take advantage of new cloud computing architectures that have emerged over the past decade to allow massive computations to be run inexpensively and efficiently. This makes operational big data workloads much easier to manage, cheaper, and faster to implement. Some NoSQL systems can provide insights into patterns and trends based on real-time data with minimal coding and without the need for data scientists and additional infrastructure.

Analytical big data includes systems like Massively Parallel Processing (MPP) database systems and MapReduce that provide analytical capabilities for retrospective and complex analysis that may touch most or all of the data. MapReduce provides a new method of analyzing data that is complementary to the capabilities provided by SQL, and a system based on MapReduce that can be scaled up from single servers to thousands of high and low end machines.

In traditional approach, an enterprise will have a computer to store and process big data. Here data will be stored in an RDBMS like Oracle Database, MS SQL Server or DB2 and sophisticated softwares can be written to interact with the database, process the required data and present it to the users for analysis purpose. This approach works well where there was less volume of data that can be accommodated by standard database servers, or up to the limit of the processor which is processing the data. But when it comes to dealing with huge amounts of data, it is really a tedious task to process such data through a traditional database server.

Google solved this problem using an algorithm called MapReduce, which was discussed earlier as analytical big data. This algorithm divides the task into small parts and assigns those parts to many computers connected over the network, and collects the results to form the final result dataset.

Apache Software Foundataion took the solution provided by Google. Apache Hadoop runs applications using this MapReduce algorithm, where the data is processed in parallel on different CPU nodes. In short, Hadoop framework is capable enough to develop applications capable of running on clusters of computers and they could perform complete statistical analysis for huge amounts of data. Hadoop is open-source software for reliable scalable, distributed computing. Many large-scale data analysis tasks have been done on this platform. A Hadoop frame-worked application works in an environment that provides distributed storage and computation across clusters of computers. Hadoop is designed to scale up from single server to thousands of machines, each offering local computation and storage.

Using a cluster with lots of computers, Hadoop provides a powerful computing ability to handle the scalability well. Hadoop consists of two layers, the data storage layer or the Hadoop Distributed File System (HDFS) and the data processing layer or the MapReduce Framework. MapReduce is a framework for the computing in Hadoop, which follows master-slave architecture. Each job is broken into map jabs and reduce jobs, which are executed in the slave nodes.

Thus Hadoop's distributed processing of large datasets across clusters of computers using simple programming models will be the best to handle RDF big data structure on the semantic web [4].

II. BIG DATA AND HADOOP MAP REDUCE

Big Data and Hadoop

Big data is a collection of data sets so large and complex which is also exceeds the processing capacity of conventional database systems. The data is too big, moves too fast, or doesn't fit the structures of our current database architectures. Big Data is typically large volume of unstructured (or semi structured) and structured data that gets created from various organized and unorganized applications, activities and channels such as emails, tweeter, web logs, Facebook, etc [5]. The main difficulties with Big Data include capture, storage, search, sharing, analysis, and visualization.
The three main terms that signify Big Data have the following properties.

- Volume: Many factors contribute towards increasing Volume streaming data and data collected from sensors etc.,
- Variety: Today data comes in all types of formats emails, video, audio, transactions etc.,
- Velocity: This means how fast the data is being produced and how fast the data needs to be processed to meet the demand.

The core of Big Data is Hadoop which is a platform for distributing computing problems across a number of servers [6]. It is first developed and released as open source by Yahoo!, it implements the MapReduce approach pioneered by Google in compiling its search indexes. Hadoop's MapReduce involves distributing a dataset among

multiple servers and operating on the data: the "map" stage. The partial results are then recombined: the "reduce" stage. To store data, Hadoop utilizes its own distributed file system, HDFS, which makes data available to multiple computing nodes [7]. Big data explosion is a result not only of increasing Internet usage by people around the world, but also the connection of billions of devices to the Internet.

Hadoop is a batch processing system for a cluster of nodes that provides the underpinnings of most Big Data analytic activities because it bundle two sets of functionality most needed to deal with large unstructured datasets namely, Distributed file system and MapReduce processing. It is a project from the Apache Software Foundation written in Java to support data intensive distributed applications [8]. Hadoop enables applications to work with thousands of nodes and petabytes of data. The inspiration comes from Google's MapReduce and google File System papers.

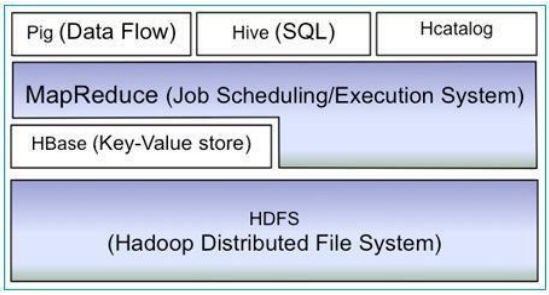

Figure 1. High level architecture of Hadoop

Pig: It is a dataflow processing (scripting) language Apache Pig is a platform for analysing large data sets that consists of a high-level language for expressing data analysis programs. The main characteristic of Pig programs is that their structure can be substantially parallelized enabling them to handle very large data sets, simple syntax and advanced built-in functionality provide an abstraction that makes development of Hadoop jobs quicker and easier to write than traditional Java MapReduce jobs.

Hive: Hive is a data warehouse infrastructure built on top of Hadoop. Hive provides tools to enable easy data summarization, ad-hoc querying and analysis of large datasets stored in Hadoop files. It provides a mechanism to put structure on this data and it also provides a simple query language called Hive QL, based on SQL, enabling users familiar with SQL to query this data.

HCatalog: It is a storage management layer for Hadoop that enables users with different data processing tools. HCatalog's table abstraction presents users with a relational view of data in the Hadoop distributed file system (HDFS)and ensures that users need not worry about where or in what format their data is stored.

MapReduce: Hadoop MapReduce is a programming model and software framework for writing applications that rapidly process vast amounts of data in parallel on large clusters of computer nodes. MapReduce uses the HDFS to access file segments and to store reduced results.

HBase: HBase is a distributed, column-oriented database. HBase uses HDFS for its underlying storage. It maps HDFS data into a database like structure and provides Java API access to this DB. It supports batch style computations using MapReduce and point queries (random reads). HBase is used in Hadoop when random, realtime read/write access is needed. Its goal is the hosting of very large tables running on top of clusters of commodity hardware.

HDFS: Hadoop Distributed File System (HDFS) is the primary storage system used by Hadoop applications. HDFS is, as its name implies, a distributed file system that provides high throughput access to application data creating multiple replicas of data blocks and distributing them on compute nodes throughout a cluster to enable reliable and rapid computations.

Core: The Hadoop core consist of a set of components and interfaces which provides access to the distributed file systems and general I/O (Serialization, Java RPC, Persistent data structures). The core components also provide "Rack Awareness", an optimization which takes into account the geographic clustering of servers, minimizing network traffic between servers in different geographic clusters.

Advantages of Hadoop

- Hadoop framework allows the user to quickly write and test distributed systems. It is efficient, and it automatic distributes the data and work across the machines and in turn, utilizes the underlying parallelism of the CPU cores.
- Hadoop does not rely on hardware to provide Fault-Tolerance and High Availability (FTHA), rather Hadoop library itself has been

designed to detect and handle failures at the application layer.

- Servers can be added or removed from the cluster dynamically and Hadoop continues to operate without interruption.
- Another big advantage of Hadoop is that apart from being open source, it is compatible on all the platforms since it is Java based.

Map Reduce Model

MapReduce is a software framework for easily writing applications which process big amounts of data in-parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner [9]. The term MapReduce actually refers to the following two different tasks that Hadoop programs perform:

- **The Map Task (Mapper):** This is the first task, which takes input data and converts it into a set of data, where individual elements are broken down into tuples (key/value pairs).
- **The Reduce Task (Reducer):** This task takes the output from a map task as input and combines those data tuples into a smaller set of tuples. The reduce task is always performed after the map task.

Typically both the input and the output are stored in a file-system. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks.

The MapReduce framework consists of a single master 'JobTracker' and one slave 'TaskTracker' per cluster-node. The master is responsible for resource management, tracking resource consumption/availability and scheduling the jobs component tasks on the slaves, monitoring them and re-executing the failed tasks. The slaves TaskTracker execute the tasks as directed by the master and provide task-status information to the master periodically.

The JobTracker is a single point of failure for the Hadoop MapReduce service which means if JobTracker goes down, all running jobs are halted.

The major advantage of MapReduce is that it is easy to scale data processing over multiple computing nodes. Under the MapReduce model, the data processing primitives are called mappers and reducers. Decomposing a data processing application into mappers and reducers is sometimes nontrivial. But, once we write an application in the MapReduce form, scaling the application to run over hundreds, thousands, or even tens of thousands of machines in a cluster is merely a configuration change. This simple scalability is what has attracted many programmers to use the MapReduce model [10].

MapReduce paradigm

MapReduce program executes in three stages, namely map stage, shuffle stage, and reduce stage.

- **Map Stage** : The map or mapper's job is to process the input data. Generally the input data is in the form of file or directory and is stored in the Hadoop file system (HDFS). The input file is passed to the mapper function line by line. The mapper processes the data and creates several small chunks of data.
- **Reduce Stage**: This stage is the combination of the 'Shuffle' stage and the 'Reduce' stage. The Reducer's job is to process the data that comes from the mapper. After processing, it produces a new set of output, which will be stored in the HDFS.

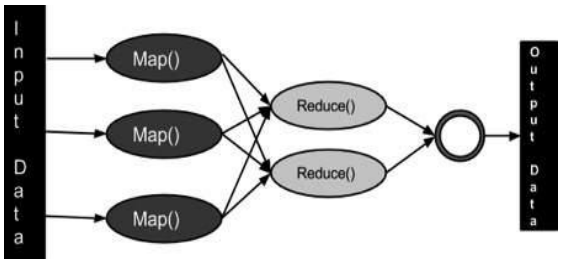

Figure 2. A MapReduce Model

During a MapReduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster. The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes. Most of the computing takes place on nodes with data on local disks that reduces the network traffic. After completion of the given tasks, the cluster collects and reduces the data to form an appropriate result, and sends it back to the Hadoop server.

Advantages of MapReduce model

- **Fault tolerance:** Data and application processing are protected against hardware failure. If a node goes down, jobs are automatically redirected to other nodes to make sure the distributed computing does not fail. Multiple copies of all data are stored automatically.
- **Flexibility:** Unlike traditional relational databases, there is no need to preprocess data before storing it.
- **Low cost:** The open-source framework is free and uses commodity hardware to store large quantities of data.
- **Scalability:** System can be updated to handle more data simply by adding nodes and also little administration is required.

III. CONCLUSION

In order to effectively handle the growing amount of available RDF data, scalable and flexible RDF data processing frameworks are needed. The emerging technologies for Big Data are Hadoop-based systems which take advantages of scalable and fault-tolerant distributed processing. Because of Google's distributed file system and MapReduce parallel model, the Hadoop-MapReduce programming paradigm has a substantial base in the Big Data community and they are still expanding. HDFS, the Hadoop Distributed File System, is a distributed file system designed to hold very large amounts of data (terabytes or even petabytes), and provide high-throughput access to these information. Ease-of-use of the MapReduce method has made the Hadoop Map Reduce technique has become more popular than any other Data Analytics techniques. Hence use of Hadoop model for RDF data processing will make the system to work easier and quicker.

REFERNCES

[1]. Stumme.G, Hotho.A, Berendt.B, "*Semantic WebMining: State of the art and future directions*", Web Semantics: Science, Services and Agents on the World Wide Web 4(2) 2006 124-143 Semantic Grid – The Convergence of Technologies.

[2] Xindong Wu, Fellow, IEEE, Xingquan Zhu, Gong-Qing Wu, and Wei Ding, *"Data Mining with Big Data"*, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 1, JANUARY 2014.

[3] SMITHA T, V. Suresh Kumar "*Application of Big Data in Data Mining*" in International Journal of Emerging Technology and Advanced Engineering Volume 3, Issue 7, July 2013.

[4] Apache Hadoop Project, http://hadoop.apache.org/, 2013.

[5] Harshawardhan S. Bhosale, Prof. Devendra P. Gadekar "*A Review Paper on Big Data and Hadoop*" in International Journal of Scientific and Research Publications, Volume 4, Issue 10, October 2014.s

[6] Dhruba Borthaku, "*The Hadoop Distributed File System: Architecture and Design"*, Retrieved from, 2010,http://hadoop.apache.org/common/.

[7] R. Taylor, "*An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics BMC bioinformatics"*,11(Suppl 12):S1, 2010.

[8] Vidyasagar S. D, A Study on "*Role of Hadoop in Information Technology era*", GRA - GLOBAL RESEARCH ANALYSIS, Volume: 2, Issue: 2, Feb 2013, ISSN No. 2277 – 8160.

[9] Mrigank Mridul, Akashdeep Khajuria, Snehasish Dutta, Kumar N "*Analysis of Bidgata using Apache Hadoop and Map Reduce*" in International Journal of Advance Research in Computer Science and Software Engineering, Volsume 4, Issue 5, May 2014.

[10] Shital Suryawanshi, Prof. V.S.Wadne, "*Big Data Mining using Map Reduce: A Survey Paper*", IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 6, Ver. VII (Nov – Dec. 2014), PP 37-40 www.iosrjournals.org.