IJRSET 2015 SPL Volume 2, Issue 9 Pages: 5-8

ISSN 2394-739X

International Journal for Research in Science Engineering and Technology

HADOOP COMPONENTS & FAULT TOLERANCE

[1] Ms. S. Shyamala, [2] Dr. M. Pushparani,
[1] MPhil Scholar, [2] Professor &Head,
[1, 2] Department of Computer Science,
[1, 2] Mother Teresa Women's University,
[1, 2] Kodaikanal, India.

Abstract:-

In this modern trend of information technology and computer science storing and processing a data is very important. Nowadays even a terabytes and petabytes of data is not enough for storing large amount of database. Hadoop is designed to store large chunks of data sets reliably. Hadoop is an open source software which supports parallel and distributed data processing. It is highly scalable compute platform. Hadoop enables users to store and process large amount of data which is not possible while using less scalable techniques. Hadoop also provide faults tolerance mechanism by which system continues to function correctly even after some components fail's working properly. Faults tolerance is mainly achieved using data duplication and making copies of same data sets in two or more data nodes. In this paper we describe the major components of Hadoop along with how fault tolerance is achieved by means of data duplication, check point and recovery[1].

Keywords: Hadoop, Fault tolerance, HDFS, Map Reduce.

1. INTRODUCTION

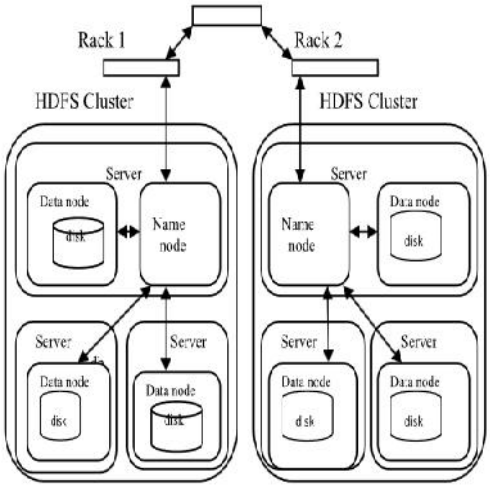
Hadoop was created by Doug Cutting, the creator of Apache Lucene, the widely used text search library. Hadoop has its origins in Apache Nutch, an open source web search engine, itself a part of the Lucene project[4].

Hadoop is an open Source large-scale data processing framework that supports distributed processing of large amount of data using simple programming models[2].Hadoop can handle all types of data including audio files, e-mails,multimedia,picture, etc.There is no limit of storing and processing data by using Hadoop.Hadoop uses computational technique named MapReduce, in which application is divided into many small fragment each of them is executed on different nodes in cluster[1]. Hadoop provides distributed file processing system that store and process large amount of data.Hadoop uses HDFS for storage purpose. HDFS is fault tolerance and provides throughput access to large data set. Hadoop is designed to efficiently process large amounts of information by connecting many computers that can work in parallel. Also one of the most important benefits of Hadoop is to limit the communication between the nodes and makes the system more reliable[1].

2. HADOOP COMPONENTS

Hadoop has two major components: HDFS (Hadoop Distributed File Systems) and Hadoop MapReduce.

Hadoop Distributed File System Each Hadoop cluster contains variety of nodes hence HDFS architecture is broadly divided into following three nodes which are, Name Node. Data Node. HDFS Clients/Edge Node[1]. Name Node This node is placed at the centre, which contains information about Hadoop file system. The main task of name node is that it records all the metadata & attributes and specific locations of files & data blocks in the data nodes. It acts as the master node as it stores all the information about the system. As name node acts as the master node it generally knows all information about allocated and replicated blocks in cluster. It also has information about the free blocks which are to be allocated next. The clients contacts to the name node for locating information ithin the file system and provides information which is newly added, modified and removed from data nodes[1].



Data Node The next type node in HDFS architecture is data node. Data node acts as slave node. Based on capacity and performance Hadoop environment may contain more than one data nodes. A data node performs two main tasks storing a block in HDFS and acts as the platform for running jobs. During the initial stage each data node performs handshakes with name node. It checks for accurate namespaces, ID if found then it connects data node to name node, and if not then it simply close the connection. Each data node keeps the current status of the blocks in its node and generates block report. Data node sends the block report to name node after every hour hence it always has updated information about the data node. During this handshaking process data node also sends heartbeats to name node after every 10 minutes, due to this action the name node knows which nodes are functioning correctly and which not. If name node doesn't receive heartbeats from data nodes it just assumes that data nodes are lost and it generates the replica of data node[1].

HDFS Clients/Edge node

HDFS Clients sometimes also called as Edge node. It acts as linker between name node and data nodes. HDFS Clients are the access points which are used by user application to use Hadoop environment. In the typical Hadoop cluster there is only one client but there are also many depending upon performance needs. When any application wants to read a file it first contacts to the name node and then receive list of data nodes which contains the required data , hence after getting that list the clients access the appropriate data node requesting the data node which can hold that file also location of replica's which is to be written. After that the name node allocates the appropriate location for that file[1].

Hadoop Map Reduce

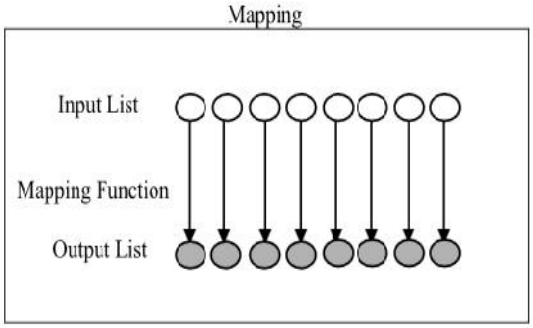
Map Reduce is a tool implemented for managing and processing huge amounts of unstructured data in parallel based on division of a large work item in smaller independent task units.Programs which are programmed to manage huge amounts of

data in parallel.To do this, load shedding is required across multiple machines[2].

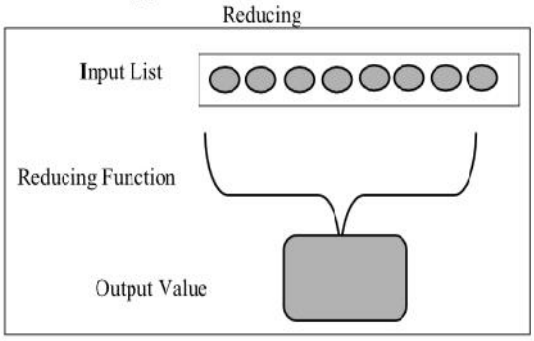
In the MAPREDUCE process the similar nature are grouped together and they are placed on the same nodes.Since MAPREDUCE elements are unchangeable in a mapper, if there is any change in input(key,value) then it will not displayed in the input files. On such circumstance the next execution with the new output values(key,value) will be taken.

List Processing- In this concept programs which are Mapreduced convert an array of data coming in as input into an array of data which is the output.Mapping and reduction functions are done when the program goes through this process.

List based Mapping- At the first instant the input data elements are received by the MAPPER and it generates the corresponding output data elements as shown below[2].

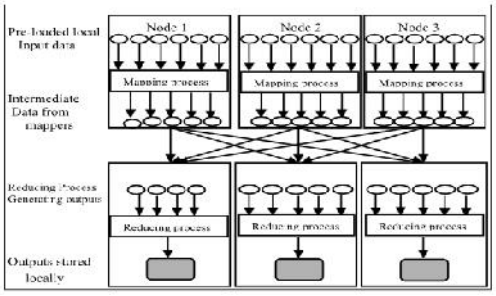


List Based Reduction- In this process the values are consolidated together. List of values are input to the reducer function from the input list and single output value is received as shown below[2]:



Combined Together- The Map Reduce Data flow consists of the mapping & reducing functions are shown below[2]:

Map Reduce Pipeline Dataflow



3. FAULT TOLERANCE IN HADOOP

HDFS and MapReduce are the ways where Hadoop provides fault tolerance mechanisms. HDFS provides storage layer of fault tolerance by replication. That is, HDFS keeps multiple replicas of each data block in several different nodes, so that if any one node is down, data could still be restored from other replicas. Hadoop MapReduce provides job level fault tolerance. That is, if a Map or Reduce task fails, the scheduler would re-assign the task to another node. Or, if a node fails, all the Map and Reduce tasks on that node would be re-scheduled to another node for re-execution[3]. Fault tolerance is one of the most important advantages of Hadoop. There are mainly two main methods which are used to produce fault tolerance in Hadoop namely Data duplication and checkpoint & recovery[1].

Data Duplication

In this method, when data copy is required or in case of fault that copy of that data are placed on several different data nodes provides us the data from another data node and provides instant recovery from failures.Though there are possibility of data inconsistency due to data duplication across various nodes, provides us instant and quick recovery from failures

hence, it is frequently used to compare checkpoint and recovery.

Checkpoint & Recovery (Rollback)

Rollback method is used to tolerate faults to some extent whereby the copy report has been saved and stored after a fixed span time interval is rollback in case of failure it starts from the last saved point by performing the transaction again. Since the rollback operation needs to go back and check for the last saved consistent stages it consumes a lot of time compared to data duplication. However it requires less additional resources.

CONCLUSION

The above analysis of Hodoop components and fault tolerance enlighten the users to store and process large amount of data and a good way of handling faults tolerance. The only support of fault tolerance on Hadoop is replication on HDFS and re-execution of failed Map or Reduce tasks. However, this increases the cost by re-execution the whole task no matter how far it proceeds. Proper skill training is also needed for achieving large scale data analysis so that we can upgrade the potential of Hadoop data management power.

REFERENCES

[1] Vishal S Patel,Pravin D.Soni- “Hadoop Skeleton &Fault Tolerance In Hadoop Clusters” paper published – Volume 2,Issue 2,Febraury 2013 International Journal of Application or Innovation in Engineering & Management(IJAIEM)

[2] Praveen Kumar, Dr.Vijay Singh Rathore,-“Efficient Capabilities of Processing of Big Data usingHadoop Map Reduce” paper published – Volume 3,issue 6, June 2014 International Journal of Advanced Research in Computer and Communication Engineering.

[3] PengHu and WeiDai –“ Enhancing Fault Tolerance based on Hadoop Cluster, International Journal of Database Theory and Application Vol.7, No.1 (2014).

[4] Tom White –“Hadoop: The Definitive Guide, Third Edition” Revision History for the :2012-01-27.