© 2016 IJSRSET | Volume 2 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Storage and Processing Speed for Knowledge from Enhanced Cloud Computing With Hadoop Frame Work : A Survey

SK. Jilani Basha, P. Anil Kumar, S. Giri Babu

Department of Computer Science & Engineering, PACE Institute of Technology & Sciences
Ongole, Andhra Pradesh, India

ABSTRACT

Cloud is a Pool of servers, all the servers are interconnected through internet, The main problem in cloud is retrieving of data (knowledge) and process that variety of data and here other problem is security for that data, Generally now a day's different types of, I mean variety of data (Structured, semi-structured and Unstructured data) is existed in the different social applications (face book).So, and another problem with historical data retrieving. These types of problems are resolved with help of hadoop frame work and Sqoop and flume tools. Sqoop is load the data from database to Hadoop (HDFS), and flume loads the data from server files to hadoop distributed file system. Storage problem is resolving with help of blocks in hadoop distributed file system and processing is resolving with help of map reduce and pig and hive and spark etc. This paper summarizes the storage and processing speed in the enhanced cloud with hadoop framework.

Keywords: Cloud Computing, Hadoop Frame Work, Infrastructure as a Service, Platform as a Service, Software as Service

I. INTRODUCTION

Now a day's the enhanced cloud computing servers and nodes are having high configurations, the hadoop framework is require a high configurations for data storing and retrieving (processing) of wanted data. Servers will have a 1 TB of hard disk capacity in present days [6]. So, the cloud server stores the image and video and test formats (content) Ex: face book. Actually data is stored in the form of rows and columns in database, it is structure data, there is no problem with structure data, sometimes applications having both image and text formats and unstructured formats, at this time facing a problem on retrieving of wanted and required query relevant data.

The rise of cloud computing made dynamic provisioning of elastic capacity on-demand possible for applications hosted on data centers [2]. This is because cloud data centers contain thousands of physical servers hosting orders of magnitude more virtual machines that are allocated on demand to users in a pay-as-you-go model. Actually some of the systems suffer with fault tolerance; those are power failures, network failures, hard and software failures (component failures) and finally metadata problems. These all are failures in normal file system. Hadoop distributed file system overcome this type of (data loss) draw backs with help of replication of data, hadoop having a replication factor is 3, hadoop stores the 512 copies maximum.

Service models:

Infrastructure as a Service (IaaS).
Platform as a Service (PaaS).
Software as Service (SaaS).

Each of these models provides a different view for users of what type of resource is available and how it can be accessed. In the IaaS model, users acquire virtual machines that run in the hardware of cloud data centers [8].

Virtual Machines (VMs) can contain any operating system and software required by users, and typically users are able to customize the VMs to their own needs. Typically, IaaS providers charge users by the time that VMs run, and the exact cost per unit of time depends on the hardware resources (memory, CPU cores, CPU speed) allocated to the VM, which users can select among different amounts offered by providers. Therefore, the views users have of the system are restricted to Operating System and above levels [5]. In the PaaS model, users are provided with an environment where applications can be deployed.

Existing Big Data ecosystem to implement advanced analytics solutions supporting Big Data-enhanced cloud computing. This include Hadoop/YARM tools (Map Reduce and other parallel programming models), Storm (stream processing), Spark use scala language, Pig and Hive (high level query languages), Mahout (high level analytics tasks), and Cassandra, HDFS- NOSQL database, Pig uses the Scripting language.

IBM provides the definition for big data in four V's. They are Volume (Bytes, MB, GB, TB, PB, EB), Velocity, Variety (Structured, Semi-structured, Unstructured), Value [3]. Hadoop is a reliable, Scalable, Platform independent, supporting the structure and object oriented programming languages.

II. METHODS AND MATERIAL

A. Architecture

The above diagram is referred from the some other reference text books of big data analytics, Contextual data suggest: figures old by our puppet criterion criteria design to apologize decisions are supplied outlandish selection sources, such as: logs immigrant the currish offensive (which may indicate impolite behavior in the corpus juries); suspicion adjacent to make noticeable of extremegoods (advocate an supplemental weight of viewable obligated in suavity or buying such products); business metrics related to expected performance parameters of the system; and facts foreigner.

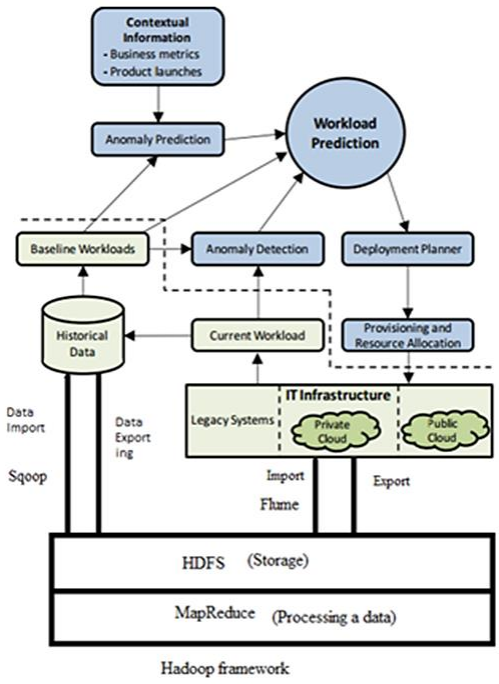

Figure 1. Architecture for importing and exporting of data with help of sqoop and flume [8].

Baseline workloads: the baseline workloads are social networks (that may indicate the sentiment of customers to a new product and may affect the input workload of the system). Formulate less the jurisprudence empirical from factual data, and assent to the determinations of fluctuations in the system input along the epoch. Such workloads supply insights on to whatever manner the liking change according to the years of the girlfriend, archaic of the week, season, months, etc. True Workload: this is the practical workload in the system in a subject scintilla and it is by-product through monitoring tools. This information is infinitely logged as reliable data for nemesis use.

2.1 Anomaly Prediction:

Alternative sources of statistics may venture Different degrees of planning, and they can be available in different formats. Modeling them as Markov models worn out settlement issues such as materials formats and dimensionality. Calculation, the assistant out are empty for enabling a precise and well-timed behave oneself of the Uncommonness Expectation coupler, consequence unequalled matter wean away exotic apposite sources are orderly for the counting and modeling; Blood of text of consistent with from the filtered Text, in addition Broad Actual Data analysis and data mining; estimation Actual of the expected workload; wariness

Determination of prediction confidence levels of failures in the system; Substitute streamer approach to be preconceived beside the malformation prediction is become absentminded the count oftheprediction deliver be favourable, therefore saunter downis sufficient years for the steadiness of the components of the system to react.

2.2 Anomaly Detection:

Since forecasts are not generally exact, and erratic circumstances might influence the workload past a level that can be anticipated, a second line of guard against loss of execution brought on by odd workloads or disappointments in the framework should be considered.

In our structure, this second line of safeguard is completed by the Anomaly Detection module. Operation of this module depends on the workload saw in a given time and standard workloads. At the point when these two estimations wander by a particular edge, an alert is activated by this module to the Workload Prediction module.

This is accomplished with abnormality discovery calculations that examine the depicted information to settle on a choice about the seriousness of the irregularity and the probability of its transiency. This is vital in light of the fact that, if the abnormality is required to acquire for a brief timeframe, it is conceivable that it stops before the earth completes its scaling procedure to handle it. Besides, if the irregularity is not serious, it is conceivable that the accessible assets can deal with it without the need of more assets. For this situation, no alert ought to be activated and the framework ought to keep its present state.

2.3 Workload Prediction:

The prior modules center in deciding examples that might prompt an expanded (or diminished) enthusiasm of clients to applications facilitated by the cloud administration supplier, an estimation of such hobby, and the danger of disappointments in the framework prompting odd conduct of the frameworks. It doesn't specifically mean a quantifiable estimation of execution of the framework in view of the startling workloads.

The Workload Prediction module completes the interpretation of watched or sudden difference in estimations to the business effect of conceivable interruptions. To accomplish this, this module measures the normal workload as far as solicitations every second along a future time window and joins this data with business sways. In this manner, the yield created by this module (and the calculations to be produced as a major aspect of its origination) is solid business measurements that have quality to chiefs of T bases.

2.4 Deployment Planning:

The Deployment Planning component of our framework is responsible for advising actionable steps related to deployment of resources in a cloud infrastructure to react to failures or anomalies in the system. Automation engine in the Provisioning and Resource Allocation module of the system executes these steps.

The tasks performed by this module are challenging as the goal of such plan is to mitigate the effect of variations in the system that disturb its correct operation. Correcting such anomalies means re-establishing a QoS level to users of the affected platform. However, enabling QoS requirements driven execution of cloud workloads during the provisioning of resources is a challenging task. This is because there is a period of waiting time between the moment resources are requested and the provision of resources by the cloud providers and the time they are actually available for workload execution. This waiting time varies according to specific providers, number of requested resources, and load on the cloud.

As our framework cannot control waiting times, this time has to be compensated by other means. Possible approaches are increasing the number of provisioned resources to speed up the workload delayed because of delays in the provisioning process or to predict earlier the resource demand albeit with low accuracy and probability. However, the first solution may not resolve the problem for most web applications because users affected by the delays are likely to abandon the access to the service, which results in loss of opportunity for revenue generation in the affected system. Another challenge for the deployment planning process concerns selection of the appropriate type of resource to be allocated. Our second approach overcomes the problem

but may be slightly more expensive due to potential over provisioning of resources. As our proposed algorithms are based on learning techniques, these methods are likely to improve their quality over the time by observing the performance of the system.

Different cloud providers have different offers in terms of combination of CPU power, number of cores, amount of RAM memory, and storage of their virtual machines. Providers also offer multiple data centers in different geographic locations. This affects the expected latency for communication and data transfer between users and resource and consequentially observed response times. Therefore, the Deployment Planning module needs to describe resource in a vendor-agnostic way, so the Provisioning and Resource Allocation module can translate the description to a concrete vendor offer once a vendor is selected.

2.5 Provisioning and Resource Allocation:

Acknowledgment of the arranging choice performed by the Deployment Planner module. Besides, distinctive blends of elements have diverse expenses. So as to meet client spending plan imperatives, the arranging calculation needs to consider the mix of assets that meet execution necessity of the assessed workload at the base expense. All the more particularly, this segment has the accompanying capacities:

- Translation of asset prerequisites from a merchant rationalist depiction to particular offers from existing cloud suppliers.
- Selection of the most suitable source(s) of assets taking into account value, inertness, asset accessibility time, and SLA.
- If conceivable, perform programmed transaction for better offers from suppliers with bargaining SLA.

2.6 Historical Data:

Historical database maintain a old data, in the above diagram historical database interact to hadoop frame work and in-between these two sqoop is useful for import and export the data from database to hadoop and flume is useful for loading the data from enhanced cloud to hadoop.

Sqoop: It is useful for import and export the data from database to hadoop.

Flume: is useful for loading the data from enhanced cloud to hadoop.

Hadoop Frame Work:

Hadoop is a open source software, it is developed by Apache Software foundation. Actually hadoop having a two type of versions in those one is hadoop-1.x and second one is hadoop-2.x. Hadoop-1.x has a some problems so go to 2.x. The problem in 1.x is single point of failure. And another thing is advantage is, hadooop-1.x having the block size is 64 MB and hadoop-2.x has a 128 MB. So, 2.x improves the through put of data.

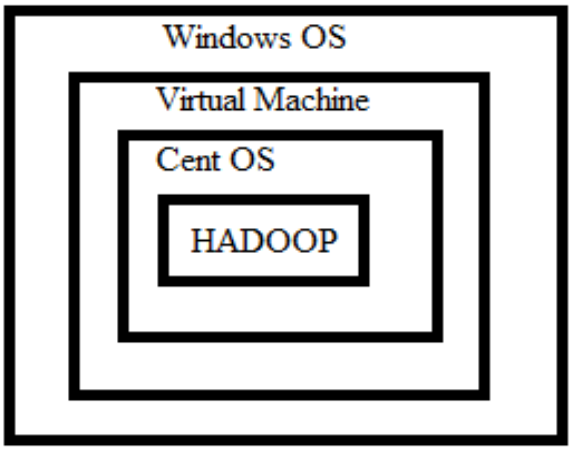

Figure 2 : Hadoop framework installations on top of Cent OS with help of virtual environment.

On top of windows directly we can't install hadoop because shell script we can't execute on windows directly. Fig 2 referred from Data Analytics and supporting distributed Architecture and operating systems text books.

Hadoop have a two core components:

- HDFS
- Map Reduce

3.1 *HDFS:*

HDFS means hadoop distributed file system. It is useful for storing the data in the form of blocks. This file system takes the data from servers and databases with respect to corresponding tools flume and sqoop. HDFS have the following process.

- *Name Node* – Storing the Metadata (data about data).
- *Data Node*- Storing the actual or original data.
- *Secondary Name Node*- It stores the back of Name Node Metadata.

Default RPC port for Name Node is **8020**.
Default RPC port for Job Tracker is **8021**.

3.2. *Map Reduce:*

Map Reduce is useful for processing the data. It is mainly having a map() and Reduce() functions. This is implementing the code in Java. And other hive implemented in HQL (Hive query language like as sql), Pig is using the Scripting language, Spark using the scala language code. These all are useful for process the data. And these are improving the process speed retrieving of wanted data from interesting patterns. Map Reduce do the distributed parallel processing.

		Process Name	RPC Port	HTTP Port
HADOOP	HDFS	Name Node	8020	50070
		Data Node	------	50075
		Secondary Name Node	------	50090
	Map Reduce	Job Tracker	8021	50030
		Task Tracker	------	50060

Table: 1. RPC and HTTP running ports for Hadoop process

Map Reduce Processes:
- *Job Tracker*- Assign the job tasks to the task tracker. And all so allot the Job ids.
- *Task Tracker*- Executes the job tasks and gives back to job tracker and again JT send report to the JT.
- HTTP means hypertext transfer protocol, these port are represented in the above table: 1.

3.3. *Challenges of hdfs:*
- Low-latency data access is not there.
- Arbitrary modifications are allowed.
- Lots of small files are an issue.
- Block is large.

Low-latency data access is not there: The response time is very less is called Low-latency. Hadoop partitions or splitting the data and stored into different replicated places, so, accessing latency is more. In hadoop-1.x HBase database solve this problem. In hadoop-2.x Spark and Drill. "Context level Indexing" is not there in hadoop. So, hadoop not allow low latency.

*Arbitrary modifications are allowed***:** Hadoop can do the 'n' number of transactions (OLAP). Hadoop performs the batch processing."Append" is provides the solution for this one. Append means adding the new data to file. It is possible in hadoop 2.x only. Write once and read n times.

Lots of small files are an issue:

Here satisfy the following terms,
- ✓ If file size is fixed, block size inversely proposal to Meta data size. (Block size is large).
- ✓ If block size is fixed, file size proposal to Meta data size. (Block size is large).

Example:

File Size	Block Size	Metadata Size
1GB	1GB	1KB
1GB	64MB	16KB
1GB	1MB	1MB

Table : 2 File size is fixed, block size inversely proposal to Meta data size

File Size	Block Size	Metadata Size
1KB	64MB	1KB
1MB	64MB	1KB
1GB	64MB	16KB

Table : 3 Block size is fixed, file size proposal to Meta data size.

Block is large: Generally Operating system Block size= 4KB or 8KB.
Seek Time: Reading the data from disk is called seek time or transfer time.

OS automatically split the data files into blocks internally but the space is miss used. But hadoop is not miss use the space of the disk.

III. RESULTS AND DISCUSSION

Master/Slave Architecture:

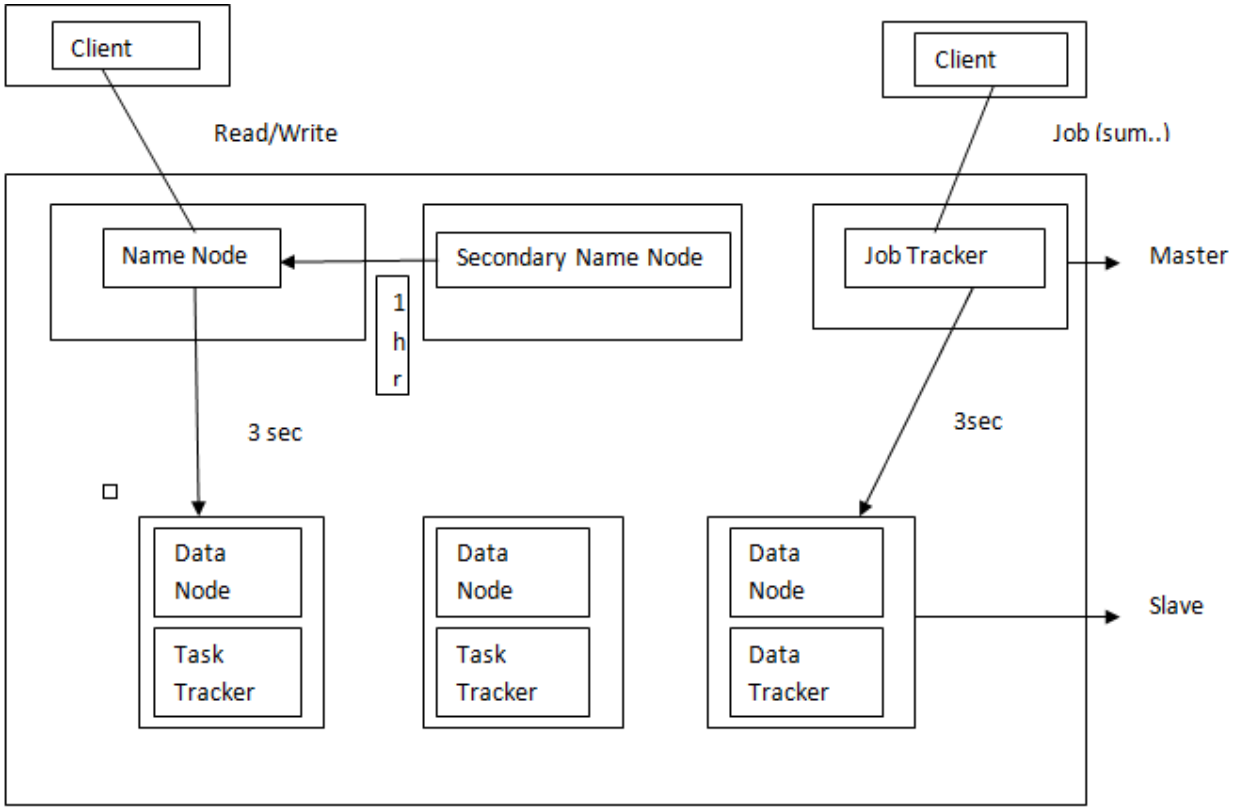

Figure 3 : Master/Slave Architecture and communication timings in between the master and slave systems.

Replication Factor: Hadoop maintain the copies of files in different nodes**.**

Default replication factor= 3.

Data loss problem is resolves with help of duplication of copies and sometimes it will give the security for the data. In the above diagram 3 shows the how data is read and write from the client system and how heart beat mechanism going on in between the master and slave for every three seconds. And storing backup of name node, these two I mean Name node is interact to secondary name node for every 1 hr.

Rack Awareness: Hadoop components are rack-aware. For example, HDFS block placement will use rack awareness for fault tolerance by placing one block replica on a different rack. This provides data availability in the event of a network switch failure or partition within the cluster.

Rack: Collection of nodes is called Rack. Here client can do read write operations.

Data Centers: Collection of racks is called Data Centers. Default rack name in hadoop is **default rack**. In here **default retries=4**.

Mainly data can be stored in the nodes on some factors, those are

- Distance
- Space Available
- Node Available
- Network speed
- RAM and Processor speed (I/O operations).

A. MAP REDUCES PROCESS:

The fig: 4 shows the how to Record reader read the input data (it may be image or video or text) and it will convert into Key and value (<K1 (line offset,

V1 (line content)>) and these value takes the maper () method, the method converts into K2, V2 and these are passed to shuffle and shot, after that it converts in to K2, list(v2). Now reducer takes those one and reduce the redundant values not for keys and converts to <K3, V3>. Finally record writer convert into output. This hadoop 1 TB of data file processed in just 62 seconds only. This is the fastness of this tool.

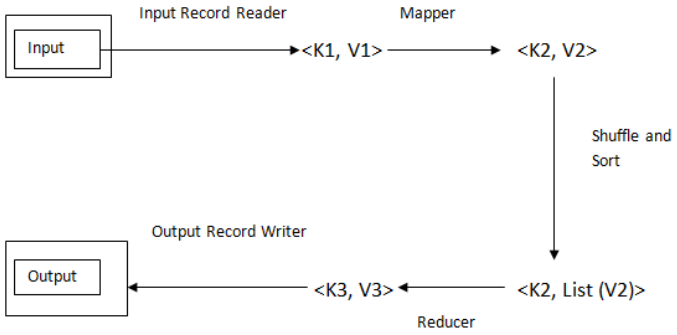

Figure 4. Map Reduce process

Cluster setup in hadoop: Hadoop supports the parallel distributed processing. So, here adding the nodes parallel in cluster. Adding the nodes is a called a commissioning and deleting the nodes from cluster is called decommissioning. But all the cluster slaves are maintained by the master node. Master/slave architecture is explained in fig 4.

B. Future Enhancement

Now a day the hadoop clustered nodes consist of high configurations may in future decreases those configuration levels for hadoop master/slave architecture. And also cloud consist of different number of clusters in cloud group because the maintenance cost is increase, in future decreases that cost and prevent the fault tolerance problems, some of tolerance problems are already prevent. In the point of processing YARN is faster. Spark is started in 2004 and Apache spark is stated in 2014. Spark is replacement of map reduce only, there is no change in HDFS. So, Apache Spark may be increases the process speed compares to map reduce. HDFS block size in hadoop 1.x is 64MB increases to 128MB in hadoop 2.x. This is improves the data storage capacity.

IV. CONCLUSION

In the cloud mainly the hadoop clustered nodes are required high configurations, but now a day's systems are built with high configurations now so, all the systems are support the framework. Storage problems are prevents and overcome with replication factor, this replication copies improve the security of data also in cloud systems. In point of processing map reduce and ache spark and coming hadoop flavors are improve the process speed.

V. REFERENCES

[1] O. Vallis, J. Hochenbaum, A. Kejariwal. A Novel Technique for Long-term Anomaly Detection in the Cloud, Proceedings of the 6th USENIX Conference on Hot Topics in Cloud Computing (HotCloud 2014), Philadelphia, USA .

[2] K. Bhaduri, K. Das, B. L. Matthews. Detecting Abnormal Machine Characteristics in Cloud Infrastructures, Proceedings of the 11th International Conference on Data Mining Workshops (ICDMW 2011), Vancouver, Canada.

[3] Y. Tan, H. Nguyen, Z. Shen, X. Gu, C. Venkatramani, D. Rajan. PREPARE: Predictive Performance Anomaly Prevention for Virtualized Cloud Systems, Proceedings of the 32nd IEEE International Conference on Distributed Computing Systems (ICDCS 2012), Macau, China.

[4] S. Islam, J. Keung, K. Lee, A. Liu, Empirical prediction models for adaptive resource provisioning in the cloud. Future Generation Computer Systems 28(1):155-162, Elsevier, 2012.

[5] T. Lu, M. Stuart, K. Tang, X. He. Clique Migration: Affinity Grouping of Virtual Machines for Inter-Cloud Live Migration, Proceedings of the 9th IEEE International Conference on Networking, Architecture, and Storage (NAS 2014), Tianjin, China.

[6] R. Buyya, C. S. Yeo, and S. Venugopal, Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities, Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC 2008), Dalian, China.

[7] R. N. Calheiros, R. Ranjan, and R. Buyya. Virtual Machine Provisioning Based on Analytical Performance and QoS in Cloud Computing Environments, Proceedings of the 40th International Conference on Parallel Processing (ICPP 2011), Taipei, Taiwan.

[8] Rajkumar Buyya, Kotagiri Ramamohanarao, Chris Leckie, Rodrigo N. Calheiros, Amir Vahid Dastjerdi1, and Steve Versteeg , Big Data Analytics-Enhanced Cloud Computing: Challenges, Architectural Elements, and Future Directions, conference paper in Proceedings of the 21st IEEE International Conference on Parallel and Distributed Systems (ICPADS 2015, IEEE Press, USA), Melbourne, Australia, December 14-17, 2015.