ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 2, Issue 2, February 2017

Hadoop: Improving Data Processing With Cloud Dataflow And Bigquery

[1] V.Trivenu, [2]P.Krishna Chaitanya
[1] [2] (Computer Science Department)
[1] [2]Balaji Institute Of Engineering And Management Studies, Rammanapalem, Nellore

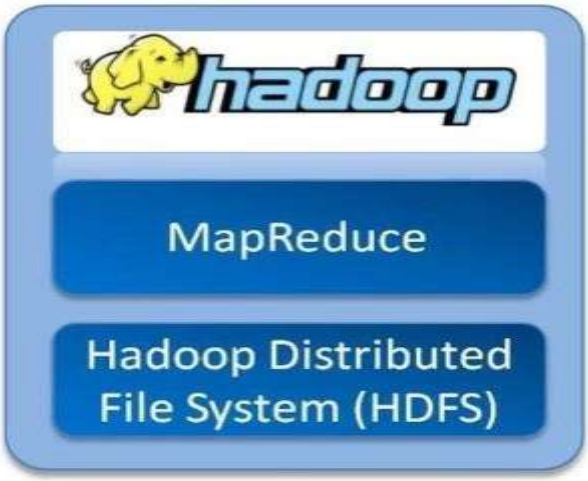
Abstract: ***Hadoop is the most popular open source implementation of HDFS and MapReduce , this powerful tools are designed for deep analysis and for faster data processing of large sets of data. Hadoop is the system that allows unstructured data to be distributed across thousands of machines forming shared clusters, and execution of Map/Reduce routines to run on the data in that cluster. Due to some flaws in the MapReduce its better to replace the MapReduce with CLOUD DATAFLOW. This cloud dataflow allows to build pipeline , monitor their execution, and transform &analyse data, all in the cloud. Inaddition to Dataflow , to fetch the data effectively we are adding bigquery.***

Keywords: Hadoop, HDFS, MapReduce, Dataflow.

I. INTRODUCTION

Hadoop is an apache project; open source framework. Hadoop Provides a distributed file system and a framework for the analysis and transformation of very large data sets using the [1] MapReduce paradigm. An important characteristic of Hadoop is the partitioning of data and computation across many (thousands) of hosts, and executing application computations in parallel close to their data. A Hadoop cluster scales computation capacity, storage capacity and IO bandwidth by simply adding commodity servers.

HDFS stores file system metadata and application data separately. MapReduce [5] is a processing technique and a program model for distributed computing based on java. The MapReduce algorithm contains two important tasks, namely Map and Reduce. Map takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs). Secondly, reduce task, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples. As the sequence of the name MapReduce implies, the reduce task is always performed after the map job.



II. HDFS AND MAPREDUCE

HDFS stands for the Hadoop Distributed File System, which is exactly what it sounds like, the file system the Hadoop platform uses to store data across many servers in a cluster. In HDFS, [9] files are broken into blocks and spread across nodes in the cluster. In addition, these blocks are replicated on different nodes to maintain data durability. This way, if one of your nodes fails another one of the live nodes still has a copy of the data.

When running Hadoop there are three main processes that manage HDFS, [6] the NameNode, the Secondary NameNode, and the DataNode. The NameNode is the master process that maintains all of the references on how exactly a file is split up in blocks across nodes in the cluster. When reading files in HDFS, a client process will contact the NameNode for this metadata and ask the corresponding DataNodes for those blocks for reading. In the current stable Hadoop release, the NameNode is a single point of failure (SPOF). This can be very dangerous because if the NameNode crashes or fails your cluster will become essentially unusable until your administrator manually recovers the NameNode. However, as Hadoop distributions from MapR, Cloudera, and HortonWorks come out with new features like federation, hot-failover, or completely replacing the NameNode all together, it is becoming less of an issue. Regardless, the Secondary NameNode daemon is an assistant process designed to take snapshots of the NameNode's metadata that can be used to recover the filesystem if you do lose the NameNode. Keep in mind this kind of recovery does require some manual human intervention. MapReduce is the data processing framework that allows developers to create applications that can take advantage of files stored in a distributed environment like HDFS. A typical MapReduce application has two functions, a Mapper and a Reducer. [4] Mappers and Reducers will be run as tasks on nodes in the cluster. The Mapper functions organize blocks of data in a way that allows the data to be aggregated and sent to Reducer functions for any kind of aggregate logic. For example, in the WordCount application the Mapper functions read blocks of text and output each word they find. If several Mappers output the same word, each of those outputs are aggregated to a single Reducer which can count the number of times it has received the same word, producing a WordCount Similar to the processes that manage HDFS, there are two processes that manage the MapReduce framework, the JobTracker and TaskTracker. The JobTracker is the master process that coordinates the Map and Reduce tasks sent across the cluster. It manages task and node failure to make sure the cluster is being used at its full potential. The TaskTrackers are slave processes that create individual Map and Reduce tasks on a node and keep the JobTracker up to date with its current status HDFS [9] and MapReduce make up the very core of the Hadoop platform. After learning about the NameNode, Secondary NameNode, DataNode,
JobTracker, and TaskTracker you should be able to make a little more sense out of some of the Hadoop buzz. Please look forward to additional blog posts where we can dive deeper into Hadoop or other interesting technologies like Storm, Hive, and HBase. Also, check out the companion webinar I did on this very same topic.

III. LIMITATIONS OF MAPREDUCE

Here are some usecases where MapReduce does not work very well [13].

1.When you need a response fast. e.g. say < few seconds (Use stream processing, CEP etc instead) [12].

2.Processing graphs

3.Complex algorithms e.g. some machine learning algorithms like SVM, and also see 13 drawfs (The Landscape of Parallel Computing Research: A View From Berkeley)

4. Iterations - when you need to process data again and again. e.g. KMeans - use Spark

5. When map phase generate too many keys. Thensorting takes for ever.

6. Joining two large data sets with complex conditions (equal case can be handled via hashing etc)

7. Stateful operations - e.g. evaluate a state machine Cascading tasks one after the other - using Hive, Big might help, but lot of overhead rereading and parsing data.

IV. WHAT IS CLOUD DATAFLOW

To say that the cloud computing market was exploding would be an understatement. In July, we heard multiple reports supporting the proclamation of cloud as the next revolution in the computing industry. The IDC claimed the cloud computing market at the close of the year would be worth $4 billion in EMEA. In the UK, 78% of organisations have "formally" adopted one or more cloud-based services. Fujitsu recently announced they've set aside $2 billion to expand their cloud portfolio. Evidently, the cloud is big business [2].

The market continues to be dominated by Amazon Web Services, with Microsoft and IBM making serious inroads. But there's one industry giant missing from this list: Google. In Q2, Microsoft's cloud infrastructure [3] revenue grew by 164%; Google lagged at only 47%. But Google have a secret weapon in their cloud portfolio, whose release may sky-rocket their market share-Google Cloud Dataflow.

ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 2, Issue 2, February 2017

In short, Cloud Dataflow allows you to build pipelines, monitor their execution, and transform & analyse data, all in the cloud. In a "sneak peek" blogpost, Google stated Cloud Dataflow will allows you to gain "actionable insights from your data while lowering operational costs without the hassles of deploying, maintaining or scaling infrastructure."

V. DATAFLOW BETTER THAN MAPREDUCE

Here are some reasons that will demonstrate that cloud dataflow is better than MapReduce. Cloud Dataflow still currently in private beta, [4] but here's an overview
of what we know so far:

1. It's multifunctional- As a generalisation, most database technologies have one speciality, like batch processing or lightning-fast analytics. Google Cloud
Dataflow counts ETL, batch processing and streaming real-time analytics amongst its capabilities.

2. It aims to address the performance issu MapReduce when building pipelines- Goo first to develop MapReduce, and the function become a core component of Hadoop. Cloud has now largely replaced MapReduce at Goo the company apparently stopped using "year ago", according to Urs Holzle, Google's Sen Technical Infrastructure.

3. It's good with big data- Hölzle stated that MapReduce performance started to sharply decline when handling multipetabyte datasets. Cloud Dataflow apparently offers much better performance on large datasets.

4. The coding model is pretty straightforward- The Google blog post describes the underlying service as "language-agnostic", but the first SDK is for Java. All datasets are represented in PCollections ("parallel collections"). It includes a "rich" library of PTransforms (parallel transforms), including ParDo (similar to Map and Reduce functions and WHERE in SQL), and GroupByKey (similar to the shuffle step of MapReduce and GROUPBY and JOIN in SQL). A starter set of these transforms can be used out of the box, including Top, Count and Mean.

5. It "evolved" from Flume and Millwheel- Flume lets you develop and run parallel pipelines for data processing. Millwheel allows you to build low- latency data-processing applications.

VI. GOOGLE IS THE MAJOR PLAYER TAPPING INTO DATAFLOW

Facebook have already developed a data flow architecture called Flux. Flux "avoids cascading affects by preventing nested updates"- simply put, Flux has a single directional data flow, meaning additional actions aren't triggered until the data layer has completely finished processing. FlumeJava, from which Cloud Dataflow evolved, is also involved the process of creating easy-to-use, efficient parallel pipelines. At Flumes core are "a couple of classes that represent immutable parallel collections, each supporting a modest number of operations for processing them in parallel. Parallel collections and their operations present a simple, high-level, uniform abstraction over different data representations and execution strategies" [9].

Many see Cloud Dataflow as a competitor to Kinesis, a managed service designed for real-time data streaming developed by industry leaders Amazon Web Services.
Kinesis allows you to write applications for processing data in real-time, and works in conjunction with other AWS products

such as Amazon Simple Storage Service (Amazon S3), Amazon DynamoDB, or Amazon Redshift [10].

VII. INFLUENCE OF DATAFLOW ON HADOOP AND MAPREDUCE

Since Cloud Dataflow is being used in place of MapReduce in the Google offices, and Google have marketed Cloud Dataflow as having "evolved" from MapReduce, many have been proclaiming the death of MapReduce, and also Hadoop, of which MapReduce is the core component.

On the subject, Ovum analyst Tony Baer told Infoworld Cloud Dataflow forms "part of an overriding trend where we are seeing an explosion of different frameworks and approaches for dissecting and analyzing big data. Where once big data processing was practically synonymous with MapReduce, you are now seeing frameworks like Spark, Storm, Giraph, and others providing alternatives that allow you to select the approach that is right for the analytic problem."

It is true MapReduce use in the decline. But that's why Hadoop 2.0 introduced YARN, which allows you to circumvent MapReduce and run multiple other applications in Hadoop which all share common cluster management. One application that's gained considerable attention is Spark; as InfoWorld states, which can perform map and reduce in-memory, making it much faster than MapReduce. Of course, such applications can run on top of Hadoop, so whilst there are now many different approaches to MapReduce, it doesn't mean Hadoop is dead. Current Hadoop users have all of their data stored on-premise, and it's unlikely that a considerable number of these users are going to migrate all of their data to the cloud to use Cloud Dataflow. In short: Hadoop is safe for now.

VIII. OVERVIEW OF BIGQUERY

BigQuery is Google's fully managed, petabyte scale, low cost enterprise data warehouse for analytics. BigQuery is serverless. There is no infrastructure to manage and you don't need a database administrator, so you can focus on analyzing data to find meaningful insights using familiar SQL. BigQuery is a powerful Big Data analytics platform used by all types of organizations, from startups to Fortune 500 companies [11].

BigQuery can scan TB in seconds and PB in minutes. Load your data from Google Cloud Storage or Google Cloud Datastore, or stream it into BigQuery to enable real- time analysis of your data. With BigQuery you can easily scale your database from GBs to PBs.

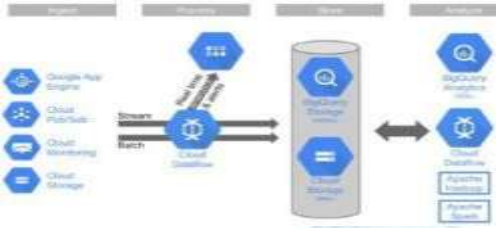
BigQuery separates the concepts of storage and compute, allowing you to scale and pay for each independently. It also gives you flexible pricing options to better suit your needs. You can either choose a pay-as-you- go model or a flat-rate monthly price for those who need cost predictability.BigQuery automatically encrypts and replicates your data to ensure security, availability and durability. You can further protect your data with strong role-based ACLs that you configure and control using our Google Cloud Identity & Access Management system.

IX. BIGQUERY FOR SCANING DATA IN HADOOP

Dataflow is designed to complement the rest of Google's existing cloud portfolio. If you're already using Google BigQuery, Dataflow will allow you to clean, prep and filter your data before it gets written to BigQuery. Dataflow can also be used to read from BigQuery if you want to join your BigQuery data with other sources. This can also be written back to BigQuery [11].

X. DATA PROCESSING IN CLOUD DATAFLOW

The following is the representation of data processing across the cloud Dataflow . along with the usage of BigQuery [13].



XI. CONCLUSION

Cloud Dataflow is being used in place of MapReduce in the Google offices, and Google have marketed Cloud Dataflow as having "evolved" from MapReduce,
many have been proclaiming the death of MapReduce, and also Hadoop, of which MapReduce is the core component.On the subject, Ovum analyst Tony Baer told Infoworld Cloud Dataflow forms "part of an overriding trend where we are seeing an explosion of different frameworks and approaches for dissecting and analyzing big data. Where once big data processing was practically synonymous with MapReduce, you are now seeing frameworks like Spark, Storm, Giraph, and others providing alternatives that allow you to select the approach that is right for the analytic problem. [7][13]" Current Hadoop users have all of their data stored on-premise,
and it's unlikely that a considerable number of these users are going to migrate all of their data to the cloud to use Cloud Dataflow. *In short: Hadoop is safe for now.*

XII. REFERENCES

[1]Apache. Welcome to Apache Hadoop.
2011. http://hadoop.apache.org/
[2]Liu Peng. Cloud computing (the 2nd edition) [M]. Beijing: Electronic Industry Press, 2011
[3]Hong Sha, Yang Shenyuan. Key technology of cloud computing and research on cloud computing model based on Hadoop [J]. Software Guide,
2010, 9(9): 9-11
[4]Jefry Dean and Sanjay Ghemwat, MapReduce:A Flexible Data Processing Tool, Communications of the ACM, Volume 53, Issuse.1, January 2010, pp
72-77.
[5]Jefry Dean and Sanjay Ghemwat,
.MapReduce: Simplified data processing on large clusters, Communications of the ACM, Volume
51 pp. 107-113, 2008
[6]Brad Brown, Michael Chui, and
James Manyika, Are you ready for the era of big data-?, McKinseyQuaterly,
Mckinsey Global Institute, October
2011.
[7]Agren, J. 2014. The materials genome and CALPHAD. Chinese Science Bulletin, 59(15), 1635-1640.
[8]"Opinion 05/2012 on Cloud Computing", 2012, [online] Available: http://ec.europa.eu/justice/data- protection/article-29/documentation/opinion- recommendation/files/2012/wp196_en. pdf.
[9]J. Baker, C. Bond, J. Corbett, J.
Furman, A. Khorlin, J. Larson, J.M. Leon, Y. Li, A. Lloyd, and V. Yushprakh: "Megastore: Providing scalable, highly available storage for interactiveservices." In: Conf.
Innovative Data Systems Research
(CIDR'11), Asilomar, California,
2011, pp. 223-234
[10]J..Tan, X. Pan, S. Kavulya, R. Gandhi, P. Narasimhan. "Kahuna: Problem Diagnosis for MapReduce-Based Cloud Computing Environments". IEEE/IFIP Network Operations and Management

Symposium (NOMS), 2010. G.
[11]Cretu, M. Budiu, M. Goldszmidt . “Hunting for problems with Artemis”. F USENIX conference on Analysis of syst logs, 2008.
[12]C. Olston, B. Reed, U. Srivastava, R.
Kumar, and A. Tomkins, “Pig Latin: a N so-Foreign Language for Data Processin in Proc. 2008 ACM SIGMOD Conferen on Management of Data, 2008, pp. 1099 1110.
[13]J. Dean and S. Ghemawat, “Mapredu Simplified Data Processing on Large Clusters,” Commun. ACM, vol. 51,
no pp. 107–113, 2008.