International Journal of Engineering Applied Sciences and Technology, 2019
Vol. 4, Issue 8, ISSN No. 2455-2143, Pages 394-396
Published Online December 2019 in IJEAST (http://www.ijeast.com)

AN EFFICIENT SCHEME TO ENSURE DATA AVAILABILITY USING BIG REPLICATE

T. Cowsalya, K. Kiruthika
Assisstant professor,
Department: Computer Science and Engineering,
SVS college of Engineering,
Coimbatore-642109, Tamil Nadu

Abstract: With the surfacing of information technologies, an overpowering amount of data and information is generated every day. Storing and handing out this enormous amount of data is named by a ubiquitous term: big data management. Cloud storage systems enhance reliability and availability of data by introducing redundancy, i.e., data replication, in the system, thereby caring the data integrity from node failures which occur commonly in any large-scale storage system. However, efficiently determining the level of redundancy, i.e., number of data replicas, is not a small task for a cloud service provider (CSP). Traditional methods, which use a fixed number of data replicas for all users regardless of the user's budget, do not achieve efficiency in terms of financial benefit of CSPs. This paper presents an efficient replication scheme called Big Replicate introduced by IBM that allows a CSP to determine the optimal number of replicas for each user depending on the user's budgetary constraint and the CSP's resource capacity while maximizing the financial profit of the CSP.

Index Terms*: *Cloud Service Provider, data replication, data reliability and data availability

I. INTRODUCTION

In digital world, data's are generated from a mixture of sources and the fast evolution from digital technologies has led to the growth of big data. It provides evolutionary breakthroughs in many fields with collection of large datasets. In general, it refers to the collection of large and difficult datasets which are tricky to process using traditional database management tools or data processing applications. These are available in structured, semi-structured, and unstructured format in petabytes and afar.

The most realistic use cases for big data involve the data availability, augmenting existing storage of data as well as allowing access to end-user employing business intelligence tools for the purpose of the sighting of data [1]. This business intelligence must be capable of connecting different big data platforms and also provide transparency of the data consumers to eliminate the requirement of custom coding. At the same time, if the number of data consumers grows, then one can provide a need to support an increasing collection of many simultaneous user accesses. This increment of demand may also spike at any time in reaction to different aspects of business process cycles [2]. It also becomes a great challenge in big data integration to make sure the right-time data availability to the data consumers.

In this study, new software called Big Replicate introduced by IBM is implemented to ensure consistent data availability.

II. BIG REPLICATE- DATA REPLICATION FOR HADOOP

Big Replicate is the world's only wide area network active transactional replication technology that delivers continuous availability, streaming backup, uninterrupted migration, hybrid cloud and burst-to-cloud, exceeding the most demanding enterprise SLAs across any combination of Hadoop distributions and cloud storage. It provides the core functionality supporting continuous availability and performance with data consistency across clusters that are any distance apart, on premises or in the cloud.

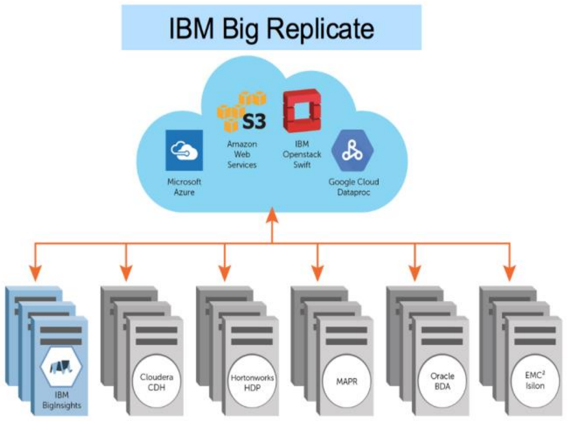


Fig 1. IBM Big Replicate- Connector

As shown in the figure 1 IBM Big Replicate is a patented technology and is built on an active-transactional replication capability that enables continuous availability and data consistency [3]. With LAN-speed performance across any combination of Hadoop clusters any distance apart, it runs on any mix of distributions and storage, both on-premise and in the cloud, enabling a whole range of enterprise use cases that can't be supported otherwise.

The proxy servers deployed with each cluster which exposes the Hadoop compatible file system API. The client apps is connected to the fusion instead of HDFS. It unifies Hadoop clusters running on a mix of distributions, versions and storage on premise and in the cloud. It provides a single virtual namespace across clusters any distance apart. It breaks down information silos [5]. It replicates data across cloud object storage and local and NFS mounted file systems. Transactional replication is typically used in server-to-server scenarios that require high throughput, including: improving scalability and availability; data warehousing and reporting; integrating data from multiple sites, integrating heterogeneous data, and offloading batch processing.

Even if there is no direct connectivity with RDBMS data sources but the data can be dumped into a file system in a CSV or similar format and this could then be mounted on HDFS with the help of IBM's Change Data Capture Tools (CDC).

III. BIG REPLICATE- PAXOS ALGORITHM

This technology has taken the Paxos algorithm and enhanced it to enable active-active replication between a variety of data sources such as Hadoop clusters, cloud environments, [network-attached storage] NAS filers and so forth. It enables continuous data access in the face of network outages, hardware failures and entire data centers going up and down, so that you get complete resilience that otherwise is impossible [4].

Paxos algorithm has the ability to maintain one-copy equivalence based on quorum agreement. What that means for multiple data sources that you want to sync is that a quorum of those data sources has to agree to any new transactions that are proposed at any one of them. The data sources have to say "yes, there is no conflict with my data, and I can accept this ordering of the transaction relative to everything else." Then once you get the quorum of those participants, the participating Hadoop clusters, databases, or whatever agreeing to that transaction, then the transaction is written by all of them effectively, what we like to say, at the same logical time.

IV. WORKING OF BIG REPLICATE

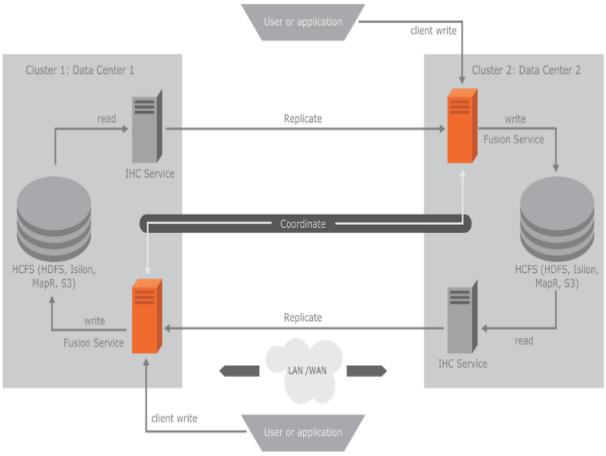


Fig 2. Big Replicate in action – Coordinating writes

Fusion service: 1 or more Fusion servers that act as a proxy for clients writing into HDFS and write replicated data into the local file system (Ref: Fusion technical paper)

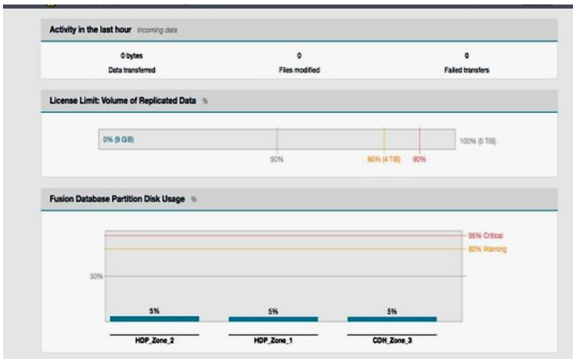
IHC service: 1 or more IHC servers that know how to read from the local underlying file system in order to send data to other clusters

Although the diagram shows two data centers, there is no limit on how many data centers you can use and you can have more than one cluster in a data center.

The labels on the lines indicate the purpose and direction of data flow: IHC reads from the file system, Fusion writes into it, and there is coordination between Fusion servers [6]. The color coding indicates coherent paths as one write comes into the HDFS and is replicated across to the other data center but it shows functions, not an accurate timeline of events. It is important to stress that active-active replication provides single copy consistency: a user or application can use the data equally from either data center. Finally, note that there are few cross-cluster network connections, which simplifies network security and management.

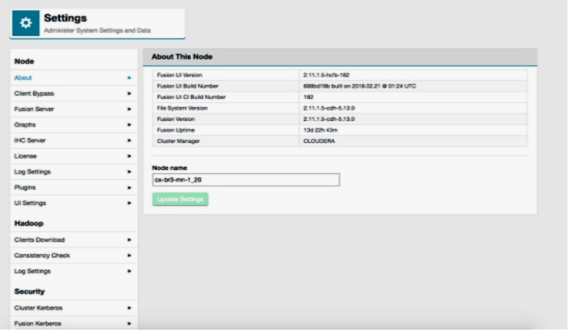
Fusion workflow

- Client makes a request to create a file
- Fusion coordinates File Open to other clusters involved (membership)
- File is added to underlying storage
- IHC server pulls data from cluster and streams to remote clusters
- Fusion coordinates File Close to other clusters involved (membership)


Activity in the last hour Incoming data

0 bytes	0	0
Data transferred	Files modified	Failed transfers

Fig 3. Dash board- Critical system status information


Settings
Administer System Settings and Data

About This Node

Fusion UI Version	2.11.1.5-hcfs-182
Fusion UI Build Number	688bd18b built on 2018.02.21 @ 01:24 UTC
Fusion UI CI Build Number	182
File System Version	2.11.1.5-cdh-5.13.0
Fusion Version	2.11.1.5-cdh-5.13.0
Fusion Uptime	13d 22h 43m
Cluster Manager	CLOUDERA

Node name: cx-br3-mn-1_20

Fig 4. Settings Management– Modify and view various big replicate settings

V. ACTIVE- ACTIVE REPLICATION

What it really enables from a practical perspective is that Big Replicate gives you the same scenario you'd have if everybody was working from one location, off of one Hadoop cluster or one database whatever it may be even though they are actually working across multiple data sources at different locations, any distance apart [8]. It gives them access to the same data, the same view of the data, read and write access to the same files just as if they were working against a single data source at a single location.

VI. BENEFITS OF BIG REPLICATE

The first is continuous availability with performance; that is, when user end up with local network speed read-and-write access to the same data in every location. Additionally, user can selectively replicate user doesn't have to replicate their entire Hadoop cluster [7]. In the case of Hadoop, replication is done at the [Hadoop Distributed File System] HDFS folder level, but the other point about this is that the data is replicated immediately as it is ingested. For example, for Spark streaming fast data kinds of applications, unlike the standard tools that you use with Hadoop for replicating data across clusters, user don't have to wait for files to be completely written and closed.

It immediately replicates the data as it is ingested. This means users are getting a kind of built-in, continuous hot backup by default because every time a transaction changes in one participating cluster, it is replicated into the others that are participating, and they can also update those same files as well. And the clusters can be on premises or in the cloud and run on any distribution. Big Replicate is agnostic to the underlying Hadoop distribution and version, unlike the tools provided by the Hadoop distribution vendors.

If a site goes down, with Big Replicate installed with each cluster, or in each cloud environment such as Big Insights on Cloud, each cluster knows the last good transaction that it processed. So when it comes back online, it is able to reach out to the other Big Replicate servers installed with the other participating clusters, grab all the transactions that it missed during the time slice it was offline, and apply them and re-sync automatically. User can eliminate the risk of human error in recovery, and it also ensures that there is no data loss.

VII. CONCLUSION

100 percent uptime, reduced costs, no vendor lock in, reduced complexity and data protection can be achieved using Big Replicate. In addition Big Replicate can lie on hybrid cloud with accelerated deployment, simplified disaster recovery, migration, upgrades, expansions etc.

VIII. ACKNOWLEDGEMENT

This work was carried out in SVS College of Engineering as a research work with the support of Department of Computer Science and Engineering.

IX. REFERENCES

[1] Kakhani M. K., Kakhani S. and Biradar S. R., (2015). Research issues in big data analytics. International Journal of Application or Innovation in Engineering & Management, (pp.228-232).

[2] Gandomi A. and Haider M. (2015). Beyond the hype: Big data concepts, methods, and analytics. International Journal of Information Management, (pp.137-144).

[3] Lynch C.,(2008). Big data: How do your data grow?, Nature, (pp.28-29).

[4] Jin X., Wah B. W., Cheng X. and Wang Y.,(2015). Significance and challenges of big data research, Big Data Research, (pp.59-64).

[5] Kitchin R., (2014). Big Data, new epistemologies and paradigm shifts. Big Data Society, (pp.1-12).

[6] Philip C. L., Chen Q. and Zhang C. Y.(2014). Data-intensive applications, challenges, techniques and technologies. A survey on big data, Information Sciences, (pp.314-347).

[7] Kambatla K., Kollias G., Kumar V. and Gram A. (2014). Trends in big data analytics. Journal of Parallel and Distributed Computing, (pp.2561-2573).

[8] Del. Rio S., Lopez V., Bentez J. M and Herrera F. (2014). On the use of mapreduce for imbalanced big data using random forest. Information Sciences, (pp.112-137).

[9] Kuo MH., Sahama T., Kushniruk A. W., Borycki E. M. and Grunwell D. K., (2014). Health big data analytics: current perspectives, challenges and potential solutions. International Journal of Big Data Intelligence, (pp.114-126).

[10] Nambiar R., Sethi A., Bhardwaj R. and Vargheese R., (2013). A look at challenges and opportunities of big data analytics in healthcare. IEEE International Conference on Big Data, (pp.17-22).

[11] Huang Z., (1997). A fast clustering algorithm to cluster very large categorical data sets in data mining. SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery.

[12] Stavros Souravlas and Angelo Sifaleras. (2011).Trends in data replication strategies : A Survey . The International Journal of Parallel, Emergent and Distributed Systems, (pp. 1-22).

Websites referred:

[1]https://www.ibm.com/in-en/marketplace/big-replicate

[2]https://www.01.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/9/897/ENUS218-279/index.html&request_locale=en