

Generate Distributed Metadata using Blockchain Technology within HDFS Environment

Shilpa K. Sangode¹, Harish K. Barapatre²

^{1,2} Department of Computer Engineering, Yadavrao Tasgaonkar Institute of Engineering and Technology
Bhivpuri Road, Karjat

Abstract - Big Data processing and analyzing is very crucial across various industries. Rapidly increasing data volume and multiple sources of data generation it has created a challenge to analyze the huge amount of data. Traditional relational database management system RDBMS cannot process the load of data generated. Hadoop Distributed File System (HDFS) is a distributed and enormously parallel processing system suited for big data whereby the storage is created on the distributed file system. Single point of failure is one of the key challenges due to the existence only one single Name Node in the existing HDFS architecture. Metadata summarizes basic information about the distribution of data blocks, block size information and replica management. Newer HDFS architecture for big data storage eliminates the concept of Name Node with the functionalities of the Name Node being stores the metadata using blockchain technology. mathematical cryptographic techniques like hashcode with the help of BlockChain Miners.

Keywords—Big data; Hadoop Distributed File System; Blockchain; Hashcode; Metadata; MINER, HDLC architecture, Single point of failure.

I. INTRODUCTION

With the fast expansion of semiconductor industry, the manufacturing process became more multifaceted and meticulous. With the advanced nature of technologies and machines data processing became more precise and huge scale. The five facets of the big data are Volume (size of data), Velocity (swiftness of data), Variety (analysis of streaming data), Veracity (data uncertainty) and Value (data quantification) [1]. Day by day the data set is growing. Conventional data processing application is not enough to dispense desired analysis and usability. Nature of data may be unstructured, semi-structured or structured. Majority of the relational database management systems (RDBMS) are process only structured data. Data are being generated and collected from variety of standalone sources in different formats. Post generation of the data it is collated together to analyze and to create insights for given problems. Businesses and institutes use intelligence to forecast future trends, for improving decision making. These insights are a valuable asset in current market. Hadoop was created to lessen the time and efforts required to query the data of numerous data formats.

Apache Hadoop is an open-source software framework. It's one of the main application is in distributed storage in Hadoop Distributed File System (HDFS); and big data set processing by means of MapReduce programming model. It

contains computer clusters made up of commodity hardware [2]. HDFS is formed of a Name Node and a cluster of Data Nodes. HDFS is must for storage and processing of terabytes or petabytes of data; especially when hard disk capacity is limited on local machines. The client data is split in multiple data nodes in a cluster which uses physical hard disk (commodity hardware's) in the nodes. Metadata contains the information about the block size, distribution of data, replica maintenance. In latest versions of Map Reduce framework it is a layer configured with block size either 64MB (default) or 128MB placed on top of prevailing operating system [3] [4].

With possible hypothesis of a HDFS architecture for big data storage which will eliminate the model of Name Node with the functionalities of the Name Node storing the metadata by means of blockchain technology. The vital role of Name Node is generating and recording metadata including the information about the position of blocks stored the size of the files, hierarchy and permissions. Additional tasks include block report from Data Nodes, maintaining and managing Data Nodes, reception of heartbeat using the blockchain technology. Blockchain stores the metadata in encrypted format. Bitcoin, the digital crypto currency uses Blockchain technology. It keeps the metadata information in a distributive fashion. Particularly, business transactions authorizations with particular mathematical cryptographic methods such as hashcode with the aid of BlockChain Miners.

II. ELEMENTS OF HDFS DESIGN

The Hadoop Distributed File System is created to gather a mammoth volume of data in several servers/clusters. Data in a Hadoop cluster is fragmented in small blocks (default is 64MB) and then dispensed throughout the clusters. Rack aware, a block placement algorithm, defines the placement of the blocks based on the replication factor, which is mostly 3 by default.

The master slave configuration decides the working principle of HDFS. A master service comprises of Name Node, secondary Name Node and job tracker besides the slave services involving Data Node and task tracker.

Master node and slave nodes are capable of communicating with each other. The slave nodes talk with each other. The job tracker can converse with the Task Tracker as illustrated in figure 1.

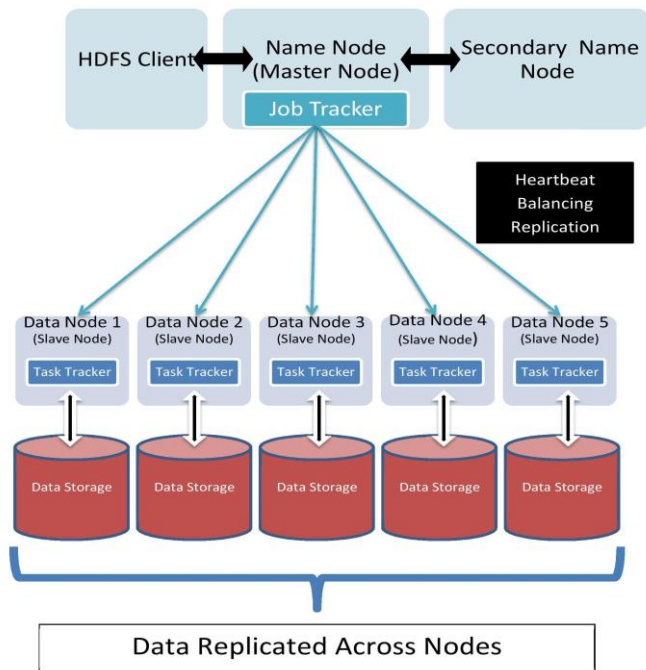


Figure 1: Elements of HDFS architecture.

NameNode: An HDFS cluster consists of an only one NameNode and it is master node. It is responsible for storage of the Meta data of files and directories. It preserves the file system tree and the metadata for all the files and directories in the tree. [3] The NameNode contains information about all the blocks of fragmented files in a cluster.

HDFS clients constantly sends the updates of metadata to namenode; when they release the write operations such as create, rename, delete, mkdir, etc. Datanodes send block report to namenode cyclically to notify it whether there are corrupted data block or not. All these modifications in metadata are stored in files, known as FsImage and EditLog resident on namenode's local disk. The FsImage comprises of a snapshot of the HDFS metadata. [5] If there is a change to metadata, it will be appended to the EditLog file. It's like a database transaction log, where alterations will not be applied to the data file immediately, however appended into a log.

When namenode fails for a definite amount of time, changes are noted in EditLog would be combined with FsImage, so that namenode could have latest report HDFS metadata. Next to combining the changes from EditLog to FsImage, namenode eliminates the old FsImage copy and exchanges it with fresher one as it has latest updated FsImage which denotes present state of HDFS and then it starts up latest EditLog. In HDFS cluster, namenode is the single point of failure for the reason that if namenode upholds the namespace and Editlog, if these records are distorted or lost, entire cluster will go down.[5] Namenode also has become the performance block of the HDFS cluster because there is only one active namenode exists in cluster.

SecondaryNameNode: In HDFS the Name Node experiences single point of failure, to recover the data some tough additions are to be performed. For the same, it is recommended to convert the Name Space Image to some

remote NFS mount device, from time to time. The secondary Name Node is combines the Name Space Image and the Edit log called checkpoints and then writes it to a file. This supports the Name Node to free the main memory occupied by the edit log till the point of last checkpoint. [4] Just in case of failure, the Hadoop administrator boots up the new Name Node and the previous data from the remote NFS mount should be recovered by hand and configure the new node.

JobTracker: JobTracker going on master node [6]. [4]JobTracker proceeding a separate node. Job tracker not usually on a Data Node and is a part of MapReduce execution in MRv1.In MRv2 job tracker got replaced by Resource Manager/Application Master. JobTracker receives the requests from the client. It then ask the Name Node to determine the location of the data. JobTracker identifies the best TaskTracker nodes to execute tasks depending on the data locality to execute a task on a given node. JobTracker observes the individual TaskTrackers and presents the complete status of the job back to the client. JobTracker process is significant to the Hadoop cluster in order to execute MapReduce. When the JobTracker is not working, HDFS will still be operational however the MapReduce execution cannot be started; the existing MapReduce jobs will be stopped.

DataNode: TaskTrackers on slave nodes [6]. Data Nodes are inexpensive commodity hardware with storage and processing capacity which are assembled within different clusters. HDFS shows a file system namespace and allows user information to be kept in files. The input file is divided into a number of files blocks and the blocks are kept in a set of Data Nodes as per the metadata report from given the Name Node. The Name Node performs operations like opening, closing, and renaming files and directories. It also governs the mapping of blocks to Data Nodes. The Data Nodes helps to read and write requests from the file system's clients. The Data Nodes also execute block creation, deletion, and replication upon command from the Name Node. [4]When a Data Node starts up, it scans across its local file system, creates a listing of all HDFS data blocks corresponding to all of these local files besides sending the report to the Name Node, which is the Block report. Heartbeat is the signal sent by all the data nodes regularly to the Name Node for retaining the relationship.

Task Tracker: TaskTracker runs upon all Data Nodes. Node Manager replaces it in MapReduce NextGen aka YARN aka MRv2. The Mapper and Reducer tasks performed, as part of Map reduce programming, on Data Nodes are administered by TaskTrackers. TaskTrackers will be allocated Mapper and Reducer tasks to be accomplished by JobTracker. [4] TaskTracker will be in continuous interaction with the JobTracker indicating the advancement of the task in execution.

III. BLOCKCHAIN

A. Background

Blockchain was initially developed for the cryptocurrency Bitcoin. It was firstly explained by Satoshi Nakamoto's in a 2008 whitepaper. Nakamoto addressed the challenge of

ownership pertaining to digital currencies plus proposed a possible way out by use of a Blockchain preceding to the settlement of transactions [8].

All the nodes within the cryptocurrency network maintain the blockchain public ledger. The blockchain warehouses all the transactions that have ever happened in the cryptocurrency system. [7] Unlike conventional centralized banking and payment systems, which have the capacity to control the system, the blockchain network does not rely on any central trusted authority or agency. Instead, trust is achieved as an emergent property from the exchanges between nodes contained by the network. Digital signatures and Cryptography mathematical models are used for generating Hash codes for every single transaction. It is believed that breaking or hacking into any transaction is highly impossible due to monitoring by all peers in the system. Transaction won't be committed unless and until by peers authorized it [3]. Any peer in the cluster can read or write transactions.

Special nodes in the [3] Cluster acts as Miners, whose securely authenticate, validate the transaction with mathematical models. The first Miner which does the mathematical cryptographic calculations and authorize the transaction will get remunerated with BitCoin called "Proof of Work" illustrated in Figure 2.

Miners: Miners are special nodes competing the transactions for validation, authentication and providing instantaneous reply to all other peers to substantiate on authentication of transaction. It uses Cryptographic mathematical models to create the necessary Hash Codes intended for the transactions based on preceding transactions plus connect the historical linked transactions altogether.[3] By confirming this "proof of work" it is remunerated.

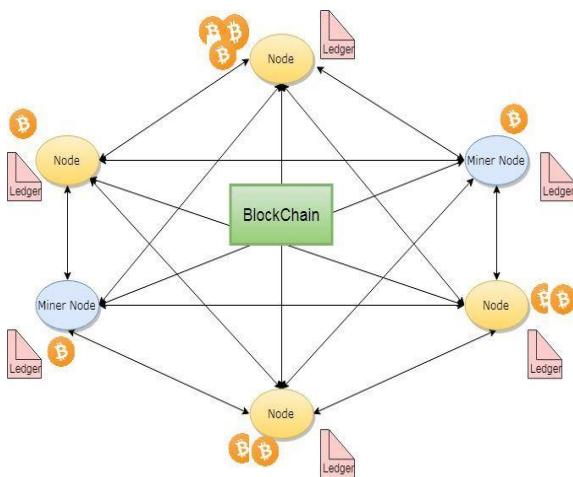


Figure 2: Blockchain Network Revolution.

B. Blockchain Hashing

Bitcoin uses hash code proof of work algorithm shown in Figure 3. The transactions are gathered as blocks. All blocks encompass latest balances of user's in the network. A blockchain is established with all the transactions (blocks)

from genesis block. A block header contains fields: Version, hash of the previous block (hashPrevBlock), root hash of the merkle tree of all transactions in the block (hashMerkleBlock), Timestamp, Nonce. The hash code algorithm repeatedly hashes the block header and raises the counter with nonce [3].

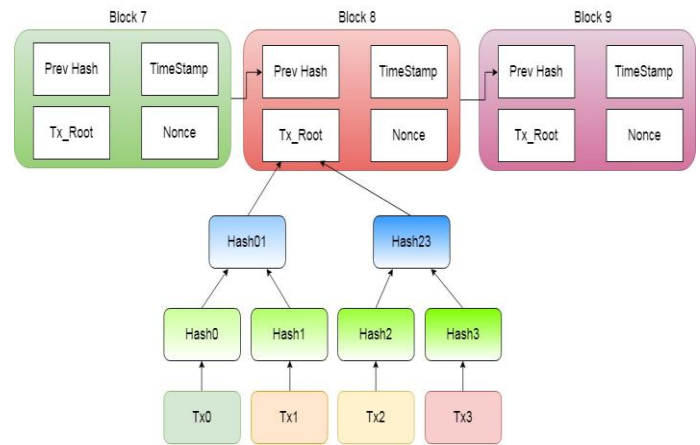


Figure 3: Blockchain Hashing.

- Version: The block version number.
- HashPrevBlock: This field obtain all the information in all the previous blocks that led to this one.
- HashMerkleRoot: This field merkle hash based on all of the transactions in the block.
- Timestamps: It is time field when Miners founds the block.
- Nonce: This field value initially set to 0 and is incremented for each hash.
- Difficulty target: This refers to difficulty of locating a new block with the help of floating point value.

With hash code algorithm, all the block headers excluding the transaction data is hashed to form the block hash. This generated block hash is used as proof that the other parts of the header have not been modified, and later this block hash is used as a reference by the succeeding block in blockchain.

IV. METHODOLOGY

In this HDFS architecture, there is a collection of Data Nodes also known as HDFS_DN data nodes are made from cluster of commodity hardware.

By eliminating the Name Node in this architecture, the design [4] becomes simpler and cheaper. However with this the metadata storage and management becomes a critical concern in HDFS design due to lack of the central authority-the Name Node. HDFS should have the capability to store,

manage, and access file metadata in reliable manner, and provide faster access to the metadata storage.

The storage of metadata is accomplished by means of spreading it throughout the Data Nodes by a master node called HDFS_MINER (one among the Data Nodes) and is stored in an encrypted form using the HDFS_Clients' public key. HDFS_MINER can be allocated to any Data Nodes depending on any arbitration mechanism in a master/slave configuration.

Blockchain technology links the data blocks in different Data Nodes in a serial manner for easy access. The block diagram of blockchain is shown in figure 4. Time stamping of attached blocks ascertains the ordering of data blocks Right back to what's called the genesis block, the very first one in the chain. The data blocks are attached with two hash values. The previous data block hash is called as H1. The current data block hash is called as H2. [4]For the first data block, H1 is from metadata and H2 computed from the current data block. Second block onwards, H1 is the previous hash value obtained from the relevant Data Nodes; H2 is calculated by the local Data Node itself. The different data blocks of the same file which are spread over various Data Nodes are connected together by means of the hash (H1) of the preceding block.[4] The current hash value (H2) stored in each block which preserves data integrity. The distributed consensus is achieved in blockchain by identifying the proper nonce value as a proof-of-work.

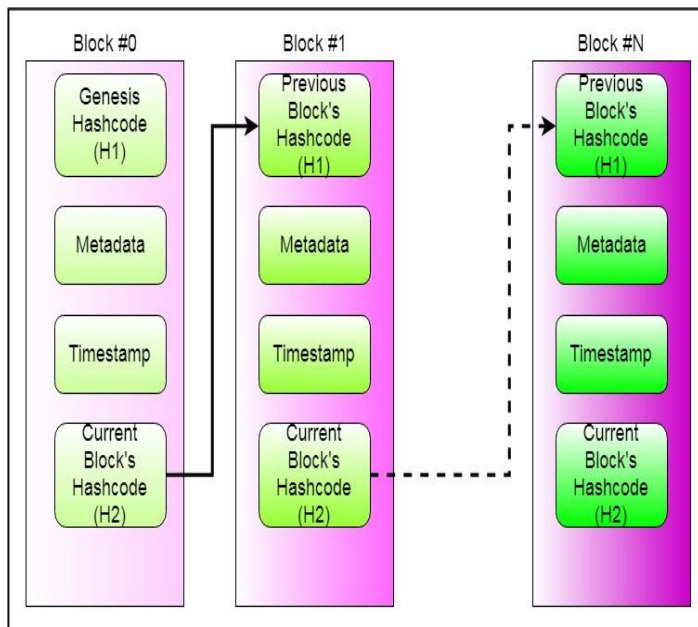


Figure 4: Blocks in the blockchain.

The following are components of HDFS

1) HDFS_Client: It requests for the storage and executes integrity check subsequent to retrieval.

2) HDFS_DN: Specialized Data Nodes whose primary goal is storage of the data blocks and a memory area earmarked for storing the metadata blocks.

3) HDFS_MINER: One or more specially delegated nodes from the Data Nodes called as MINER nodes. They help in metadata creation along with data block storage.

A key feature of this architecture is to store metadata securely throughout the cluster of Data Nodes in an encrypted form. In the HDFS architecture, the concept of Name Node is not employed. The simple block diagram of new HDFS architecture with using blockchain technology is shown in figure 5.

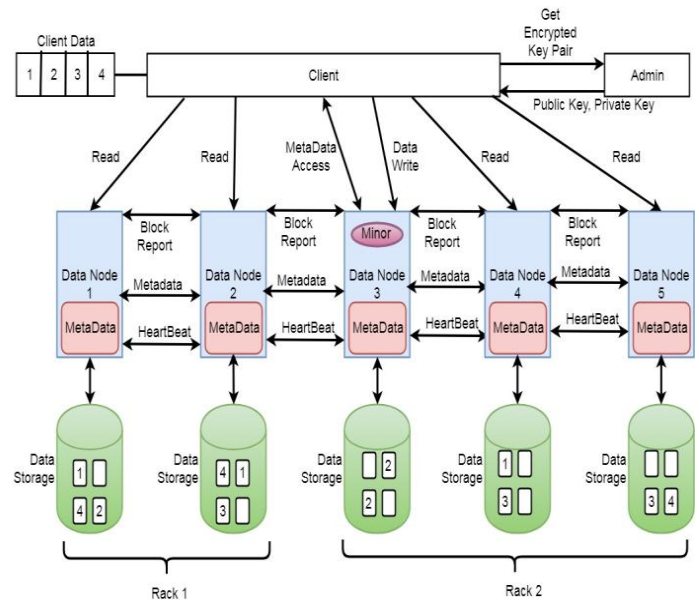


Figure 5: HDFS architecture with using blockchain Technology.

Working procedure HDFS architecture with blockchain technology

1. HDFS_Client requests the Admin to acquire the key pair for encrypting the corresponding metadata as well as data blocks, as per requirement.

2. By means of Token passing method or any alternate Arbitration mechanism, selects one SHDFS_MINER from the data nodes.

3. On receipt the key pair, HDFS_Client requests the HDFS_MINER for data storage.

4. Data is first written to a HDFS_MINER when a client is writing data to an HDFS file. Then HDFS_MINER creates the metadata .This contains the details of Data Nodes in which the data blocks and its replica blocks to be stored .The metadata will be stored in a temporary storage. Hadoop configuration specifies the default block size and replication factor. Block size, the number of replicas, and the replication factor for a specific file at its creation time are specified by an application.

5. Heartbeats, failure report and Block report

- All peer Data Nodes sends a heartbeat signal mutually to recognize the failure nodes within the cluster. If a particular node doesn't receive the heart beat within the

specified time period, peer node will send a failure report to the MINER and the MINER will construct the another metadata.

- The Data Nodes send Block report following successful storage of the data blocks to the MINER.

6. Once all the block reports are received in MINER, it will send the replica of metadata encrypted by means of the HDFS_Client's public key to all the related Data Nodes.

- The replica management is done by Rack Awareness standard rules. The replica copies are limited to 3; two replicas are positioned on one rack and the third one is in a different rack.

7. Blockchain creation: Each block of the blockchain is composed of two hash values (H1, H2) and the data fragment (64MB/128MB). The hash of the previous block stores the first hash value (H1); second hash (H2) stores the hash computed from the present data fragment. The first hash value (H1) is used to attach different data fragments in different HDFS_DNs.

8. HDFS_Client can read the metadata from HDFS_MINER stored in encrypted form. This can be decrypted using its private key provided by admin and reading data from the corresponding Data Node. After getting the first block of data, it is simple to access the successive blocks within the blockchain.

9. Data Nodes, HDFS_MINER failure:

- If the Data Node fails, it is detected from the failure report to the HDFS_MINER and it modifies the metadata.

- If the MINER node fails, another node will act as HDFS_MINER through arbitration mechanism.

V. CONCLUSION

The current HDFS architecture has limitation from single point of failure and a contemporary architecture was suggested for small sized cluster of nodes to ease the trouble of single point of failure. Also the secondary name node is utilized to create checkpoints to the remote NFS mounted device. Both are maintained employing expensive reliable hardware. In the new architecture both expensive hardware is being removed and consequently making it a cost effective strategy. The blockchain stores replicas of metadata on a rack by rack basis with one Data Node / rack and cost of maintaining blockchain. The application of blockchain technology, there is no point of recovery due to the presence of several replicas of the metadata being spread throughout one Data Node per rack. Hence the Name node and secondary name node is not essential in the proposed HDFS architecture.

REFERENCES

[1] Sheng Kang¹, Wei-Ting Kary Chien¹, Jun Gang Yang² "A Study for Big-Data (Hadoop) Application in Semiconductor Manufacturing", Proceedings of the 2016 IEEE IEEM, p 1893-1897. [2] Roscoe Lin, Veda Shankar "Performance Study of

Ceph Storage with Intel Cache Acceleration Software", 2017 IEEE 4th International Conference on Cyber Security and Cloud Computing, p 10-12.

[3] Lakshmana Kumar Yeddu, Shireesha Yeddu "Hadoop: Bitcoin-BlockChain - A New Era Needed In Distributed Computing", International Journal of Computer Applications (0975 - 8887) Volume 154 - No.6, November 2016.

[4] Deepa S. Kumar, M. Abdul Rahman "Simplified HDFS Architecture with Blockchain Distribution of Metadata", International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 21 (2017) pp. 11374-11382.

[5] Zhanye Wang, Dongsheng Wang "NCluster: Using Multiple Active Namenodes to Achieve High Availability for HDFS", 2013 IEEE International Conference on High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing, p2291-2297. [6] Shiori KURAZUMI, Tomoaki TSUMURA, Shoichi SAITO, and Hiroshi MATSUO "Dynamic processing slots scheduling for I/O intensive jobs of Hadoop MapReduce" IEEE 2012 Third International Conference on Networking and Computing.

[7] Xiwei Xu, Cesare Pautasso, Liming Zhu, Vincent Gramoli, Alexander Ponomarev, Shiping Chen "The Blockchain as a Software Connector", 2016 13th Working IEEE/IFIP Conference on Software Architecture.

[8] Matthias Mettler "Blockchain Technology in Healthcare", 2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)

[9] Sahar Mirzayi, Mohammad Mehrzad "Bitcoin, An SWOT Analysis", 7th International Conference on Computer and Knowledge Engineering (ICCKE 2017), October 26-27 2017, Ferdowsi University of Mashhad, pp 205-510.

[10] Nazri Abdullah, Anne Håkansson, Esmiralda Moradian "Blockchain based Approach to Enhance Big Data Authentication in Distributed Environment", 2017 IEEE, pp 888-892.