

Load Balancing in Distributed File Systems

Abhishek Sharma, Prashansa Prachi, Rajesh Babu Chandran

Abstract: With the development of the system and expanding clients, and also rising new advancements, overseeing customary systems and adjusting heap of cloud servers to meet levels of popularity is a rising test. In this paper, a completely conveyed load adjusting calculation is displayed to adapt to the heap unevenness issue. The proposed framework utilizes Software-Define- Networking (SDN) rather than customary load balancers, which permits better system administration, control and arrangement authorization. The normal outcomes may show that the proposition is equivalent with the current unified methodology and it may impressively outflank the earlier dispersed calculation as far as load unevenness factor, development cost, and algorithmic overhead.

I. INTRODUCTION

The ongoing development of cloud frameworks has motivated merchants to build up their foundation and relocate their items to cloud, which requires various servers that may increment all the time to serve different sorts of administrations.

Appropriated record frameworks are key building hinders for distributed computing applications dependent on the Map Reduce programming worldview.

In such document frameworks, hubs at the same time serve processing and capacity works; a record is divided into various lumps allotted in particular hubs with the goal that Map Reduce errands can be performed in parallel over the hubs. Be that as it may, in a distributed computing condition, hubs might be updated, supplanted, and included the framework.

Load Balancing is a method to convey the heap onto distinctive assets with the end goal to stay away from over-burden on any of the assets. A portion of the objectives of load adjusting are limiting reaction time, improving movement and amplifying throughput.

Ordinary systems don't have a worldwide perspective of the system, thus utilizing burden adjusting components in customary systems isn't exact. In any case, SDN strategies are more exact and have superior.

II. LITERATURE SURVEY:

[1] A network in a laptop:

This paper aims to support collaborative network research

by getting the maximum value of mininet which enables self contained SDN prototypes. Topology discovery algorithms which we can use with mininet to implement different

topologies to reduce errors, experiment and validate a fix. The author has performed extensive research on the advantages of using mininet for creating new SDNs because of its efficient use of time and resource, interactive prototyping and scalability.

[2] Map Reduce: Simple data processing on large clusters: Objective of this paper is implementing map reduce programming model for processing and generating large datasets over distributed systems. The author proposed to restrict the programming model to easily parallelize and distribute computations. Optimising the algorithm for minimizing the data sent across networks.

[3] The Google file system:

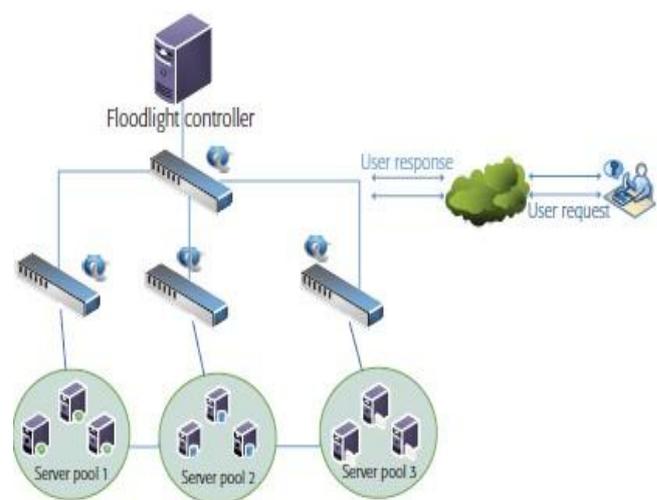
It demonstrates the qualities vital for processing massive scale records processing workloads on commodity hardware. Some of the layout patterns used are namespace management and locking, garbage collection, fault tolerance and analysis.

[4] Load balancing in data center networks:

The goal is to perform load balancing, allow bandwidth maximization, utilization and saving electricity for information center networks.

Despite the fact that there are still a number of studies issues and demanding situations in load balancing, it is nevertheless promising to approach these demanding situations by means of new developing technologies.

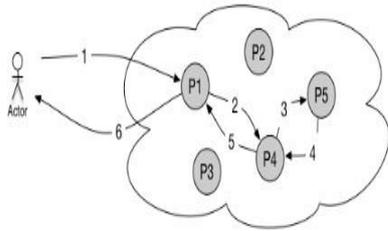
III. PROPOSED SYSTEM:



The modules used are: Chunk Creation, DHT Formulation, Load Balancing Algorithm and Replica Management.

Revised Manuscript Received on September 22, 2019.

Abhishek Sharma,
Prashansa Prachi,
Rajesh Babu Chandran



- | | |
|---|---|
| <ol style="list-style-type: none"> 1. STORE "MyKey" / "My Value" 2. I'm not responsible for "MyKey", P4 is closer, so ask P4 3. I'm not responsible for "MyKey", P5 is closer, so ask P5 4. OK, value is stored 5. OK, value is stored 6. OK, value is stored | <ol style="list-style-type: none"> 1. GET "MyKey" 2. I'm not responsible for "MyKey", P4 is closer, so ask P4 3. I'm not responsible for "MyKey", P5 is closer, so ask P5 4. OK, here is "My Value" 5. OK, passing back "My Value" 6. OK, passing back "My Value" |
|---|---|

- 4) B Amutha, Swasti Saxena, Abhinaba Das, Changing mechanisms of enterprise security (comparing beyond corp with prevalent network security mechanisms).
- 5) Jiao Zhang and F. Richard Yu, "Load Balancing in Data Center Networks: A Survey"
- 6) T Rashmi et al., "Load Balancing as a Service in Open-stack-Liberty" Int'l J. Scientific And Technology Research, Vol 4, no. 8, 2015
- 7) B.P Rimal and M. Maeir, "Workflow Scheduling in Multitenant Cloud Computing Environments" IEEE Trans. Parallel and Distributed Systems, Vol 28, no. 1, 2017, pp.290-304
- 8) F. Ertam and E. Avci, "Classification with Intelligent Systems for Internet Traffic in Enterprise Networks," Int'l. J. Computing, Communications & Instrumentation Eng., vol. 3, 2016, pp. 9–15.
- 9) X. Gao et al., "Traffic Load Balancing Schemes for Devolved Controllers in Mega Data Centers,".

Each chunk server node is estimated to be under loaded or overloaded. A node is mild if the number of chunks it hosts is lesser than the brink. Every node contacts some randomly selected nodes and constructs a vector V. A vector consists of entries and each access includes the identification, status of current load and network address of a node. Some of the advantages include- nodes take more loads, the speed and consistency is maintained.

IV. SCOPE

The proposed mechanism can lessen the average response time, reply time and request in line with seconds compared to the present day schemes. It is also determined that the centralized schemes can appropriately accumulate worldwide congestion data and stability load close to optimally.

V. CONCLUSION

As the loads on cloud servers show rapid increases, it is very important to find new and more efficient ways to balance the loads. The proposed scheme uses SDN algorithms, which lets in higher control, network management and enforcement of policy and as some of the advantages- nodes can take more loads, the speed and consistency is maintained.

VI. FUTURE WORK

There exists a lot of new algorithms and ways to be implemented in future. One major work would be to develop a system with GUI for server based load balancing.

REFERENCES

- 1) Ashutosh Phatak ; Raturaj Kadikar ; K.Vijayan ; B.Amutha(20 18, April). Performance Analysis of Firewall Based on SDN and OpenFlow. In 2018 International Conference on Communication and Signal Processing (ICCSP) (pp. 0611-0615). IEEE.
- 2) Prathamesh Purohit ; Raturaj Kadikar ; M. Susila ; B. Amutha(2018, April). Study of Service Chain Optimization in Cloud Environment. In 2018 International Conference on Communication and Signal Processing (ICCSP) (pp. 0605- 0610). IEEE.
- 3) B.Amutha, Bhavani Ghanta, Karthick Nanamaraman, Manickavasagam Balasubramanian, M. (2015). ECOSENSE: an energy consumption protocol for wireless sensor networks. Procedia Computer Science, 57, 1160-1170

