Load Rebalancing Using Peers in Clouds

T.Rupa Anusha [1], S.Siva Sankar Rao [2]
[1] M.Tech Student, Aurora's Technological and Research Institute, JNTU, Hyderabad
[2] Associate Professor, Aurora's Technological and Research Institute, JNTU, Hyderabad

Abstract: System based on the distributed scenario plays a major role for the implementation of the applications related to the cloud. Distributed file system is a major phenomenon in the cloud environment. It is an application oriented paradigm program which is based on the map reduces oriented strategy. A proper imbalance in the system takes place due to the automatic generation of the data in the form of files and their creation followed by the appending and deleting of peers in a respective aspect. The master node has to rebalance in such conditions but as the system size increases the master node unable handle load may fail. Also balancing among them is also difficult without any global information about the system as the global knowledge is with the master node. So here we conclude that this proper imbalance is the major problem related to the present system and it is solved by first making use of mapping and reduce paradigm and later when imbalance occurs then the nodes acquire knowledge about the other nodes by making use of gossip based aggregation protocol. The peers instead of depending on the central node for balancing, tries to balance among themselves. Here instead of nodes peers are used.

Keywords: Computation based cloud, Network sharing, Balancing load, File system based distribution, rebalancing scenario respectively.

1. Introduction

A technology based on the compelling based strategy is termed as the computation based on cloud which provides services such as hardware and software that are used as computing resources. There is a huge advancement in the technology related to this particular aspect [1].

This may be due to the advancement in the internet based facility. Here we conclude that the advancement in the field of internet can be termed as the computations based on the cloud.

Without any management oriented strategy there is an effective method for the allocation of the resources based on the choice of the user in a well efficient fashion. Many of the technologies related to the cloud oriented phenomena includes programming of the map reduced strategy, System based on the distributed file oriented scenario and virtualization etc. [2] [3].

This is rather a scalable phenomenon where they arbitrarily rely on the system oriented aspects respectively. System based on the file distribution based strategy is main role for the implementation of the applications related to the computations based on the cloud oriented aspect. Where an automatic analysis of the system takes place with respect to the nodes oriented strategy in a well efficient manner.

Here operations takes place in a computation reduced fashion where the effective outcome of the system is in the hands of the proper functionality based criteria followed by the proper analysis of the system where there is a complete tracking and the monitoring of the system in a very effective manner and also the replacement of the nonfunctioning peers plays a major role for the good functioning of the system.

Here the main strategy or the major functionality of the node based system is the efficient storage which is followed by the proper functionality of the computation based phenomena in a well respective aspect.

Here there is a simultaneous processing of data take place where there is an accurate partitioning of the data followed by the proper allocation based scenario and the analogous process of the scenario [4] [5].

In google and hadoop systems the balancing is done by the centralized system though these systems are so far good with respect to the performance oriented concepts they still face so many problems like choosing master node and balancing the nodes which are also very important.

Here though balancing the load at the beginning is done by master node later balancing occurring due to deletion or addition nodes is done among the peers itself without depending on the master node thus preventing the master node from further failures.

2. Load Balancing Problem

The load balancing in a cloud environment in the existing file systems like google file system and hadoop file system is mainly done by the master node or centralized node. The centralized node has to first divide the file into file chunks and has to distribute the file chunks to the chunk servers using map reduce paradigm.

Here in cloud atmosphere as nodes are replaced, deleted and also added and also file chunks are also added, updated or deleted so balancing should be done in a cloud whenever there is an imbalance.

This balancing is done by the centralized node. As the load increases in order to balance lot new nodes are added and thus the scalability increases. The centralized node cannot handle the increasing load and may fail thus increasing the delay in processing the request and thus failing in handling the requests. In such situations to provide better solution a load rebalancing methodology is used in the proposal.

3. Methodology

In this paper a method is designed with a well effective strategy where it is used for the implementation of the system in a well efficient manner [6]. Here the present designed method is described by the below block diagram in a well efficient manner.

Here the method is effective and efficient in terms of the performance based strategy and completely overcome the drawbacks of the several previous existing techniques and control the performance degradation of the present system based aspect followed by the accurate improve in the overall system in terms of the analysis [7] [8].Here in the present designed method there is an effective study of the problem related to the load unbalancing.

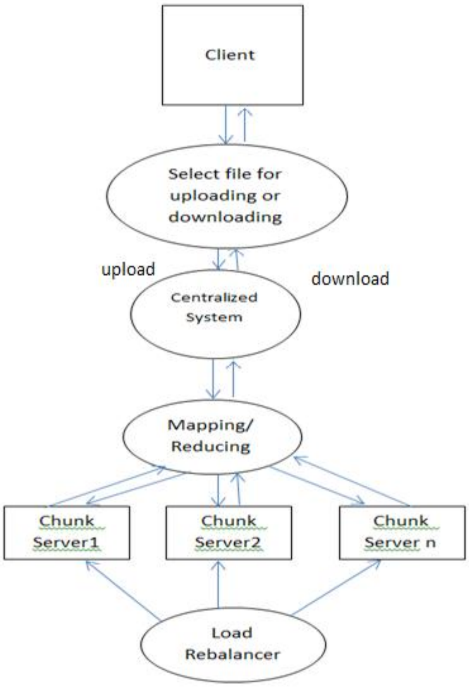

Fig 1: Block Diagram of the Methodology

In the above diagram explains the system we are developing in a detailed fashion. Here as we can observe the nodes in the system are client, centralized system and the nodes.

The functionality of the client is to first selecting a file after which he can upload a file and also he can download a file which is previously uploaded by him.

The centralized system on the other hand split the uploaded file into pieces of file just performs the mapping and reduce paradigm. The mapping and reducing technique has two tasks one is mapping and the other is reducing. The mapping task functionality is to distribute the pieces of file among the nodes. Later if the client wants to download the uploaded file all the pieces of file are reduced using the reduction task into a single file and is available for download.

The nodes are at first balanced by the centralized node. In clouds there are frequent changes which may create certain sort of imbalance among them. In this case instead of the depending on central node to solve the unbalancing situation the nodes it tries to solve the imbalance by using a rebalancing technique.

The distributed phenomena play a major role in the implementation analysis. It is mainly applied for the purpose of the data integration related to the large scale oriented strategy of the cloud oriented strategy in a well efficient manner [9]. Here the main and the primary objective of the system is the proper allocation of the data based on the aspects of the chunks oriented phenomena. Here we conclude that the present system is effective and efficient in terms of the performance based criteria followed by the accurate analysis with overall system perspective [10].

4. Rebalancing Technique

Whenever an imbalance occurs due to modification in the cloud the rebalancing technique is used. In this instead of the central node handling the imbalance the nodes try to balance among themselves.

Here least loaded node tries to balance the heavier or overloaded node. Without proper information about the capacity of other nodes in the cloud the balancing is difficult.

This is due to the fact that how can least node find a heavy node. In order to solve the problem a gossip based technique is used in which a node actually maintains information about some randomly selected nodes. In this case load information is collected. It then sorts the loads and then identifies its position among the nodes and accordingly identifies the node it has to balance.

Whenever a node leaves the system it releases its load to its successive node and then leaves the cloud. Similarly when a new node gets added to the system then it is balanced sharing the load of heavily loaded node.

But this existing process also creates problems. When a new node is added the preceding node load is transferred to the new node. Here the problem is what happens if the new node is overloaded. Later the preceded node joins as successor to heavy node and tries to balance it. This is a time taking process and yet low in performance.

When a node is deleted its load is transferred to successor. In such a situation the successor may overload. Thus the data may be lost or the system may fail. This system instead of providing a solution it causes overhead which is low in performance and security criteria.

The solution to this problem is to replace the client server paradigm with peer – peer distributed paradigm. Here every node can act as client or server. It can send request and wait for result and also process a request and send solution.

Here a peer can perform rebalancing among other peers by attaining global knowledge by itself. It takes the functionality of a server during rebalancing. When distributing the load they act as clients. Here a solution is provided to the complicated situation without reducing the performance.

Whenever a new node is added the remaining peers try to balance it by again distributing among themselves using mapping and reduce paradigm.

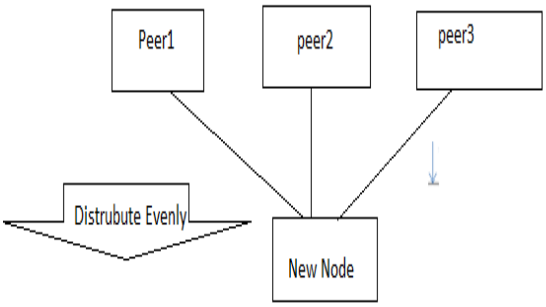

Fig 2: Rebalancing when New Node is added

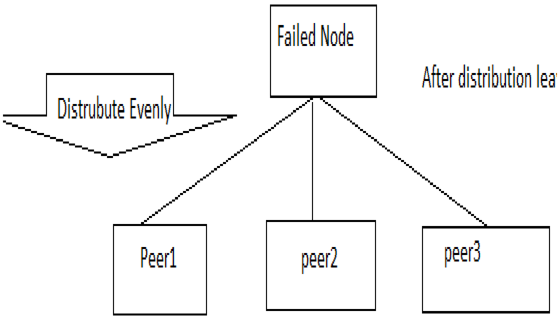

Fig 3: Rebalancing when a node is deleted

Whenever a new node fails before getting removed from the cloud the node distributed its load to the remaining peers and then leaves the system.

5. Expected Result

The expected result should be an environment of cloud in which all the nodes are perfectly balanced without any imbalances that is neither under loaded nor over loaded.

The process of rebalancing takes place until the system is completely balanced. The rebalancing technique is proposed to balance them.

The file uploaded by the file should be saved without any pieces of files getting deleted and the client should be able to download the whole file by combining all the pieces.

The system developed is therefore to avoid the huge load on the master. Which is achieved effectively and therefore reducing the master node break down as the nodes in the system increases.

The peer environment works effectively without affecting the actual performance and improving the system performance effectively.

The expected result is a completely balanced system with availability of files uploaded and also downloading the files uploaded by the user.

6. Conclusion

In this paper a method is designed with an effective strategy oriented framework. In order to improve the performance of the entire system accurate analysis is performed in a well defined fashion. There is a huge challenge for the present method where it is implemented to balance the load and to improve the design oriented consequences in a well effective fashion. Here an algorithm is designed based on the balancing of the load followed by the dynamic large scale implementation over the cloud. The cloud based distributions have been presented in an effective manner. Here the main task of the system is to control the imbalance of the nodes by the balancing based phenomena and also reducing the cost which is a huge advantage. With the help of the above network oriented strategy there is a reduction in the complexity of the system followed by the property of the peer based architecture. Here a huge analysis of the system takes place with respect to the ideal condition followed by the work based overload condition and find the differentiation in order to accurately improve the performance of the system.

7. References

[1] S. Ghemawat, H. Gobioff, and S.-T. Leung, “The Google File System,”in Proc. 19th ACM Symp. Operating Systems Principles (SOSP’03), Oct.2003, pp. 29–43.

[2] H. Abu-Libdeh, P. Costa, A. Rowstron, G. O’Shea, and A. Donnelly,“Symbiotic Routing in Future Data Centers,” in Proc. ACM SIGCOMM’10, Aug. 2010, pp. 51–62.

[3] I. Raicu, I. T. Foster, and P. Beckman, “Making a Case for DistributedFile systems at Exascale,” in Proc. 3rd Int’l Workshop Large-Scale Systemand Application Performance (LSAP’11), June 2011, pp. 11–18.

[4] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, andS. Lu, “BCube: A High Performance, Server-Centric Network Architecturefor Modular Data Centers,” in Proc. ACM SIGCOMM’09, Aug. 2009, pp.63–74.

[5] S. Surana, B. Godfrey, K. Lakshminarayanan, R. Karp, and I. Stoica,“Load Balancing in Dynamic Structured P2P Systems,” PerformanceEvaluation, vol. 63, no. 6, pp. 217–240, Mar. 2006.

[6] M. Jelasity, A. Montresor, and O. Babaoglu, “Gossip-Based Aggregationin Large Dynamic Networks,” ACM Trans. Comput. Syst., vol. 23, no. 3,pp. 219–252, Aug. 2005.

[7] H.-C. Hsiao, H. Liao, S.-S.Chen, and K.-C. Huang, “Load Balance withImperfect Information in Structured Peer-to-Peer Systems,” IEEE Trans.Parallel Distrib. Syst., vol. 22, no. 4, pp. 634–649, Apr. 2011.

[8] Q. H. Vu, B. C. Ooi, M. Rinard, and K.-L. Tan, “Histogram-Based GlobalLoad Balancing in Structured Peer-to-Peer Systems,” IEEE Trans. Knowl.Data Eng., vol. 21, no. 4, pp. 595–608, Apr. 2009.

[9] Y. Zhu and Y. Hu, “Efficient, Proximity-Aware Load Balancing for DHTBasedP2P Systems,” IEEE Trans. Parallel Distrib. Syst., vol. 16, no. 4,pp. 349–361, Apr. 2005.

[10] H. Shen and C.-Z.Xu, “Locality-Aware and Churn-Resilient Load BalancingAlgorithms in Structured P2P Networks,” IEEE Trans. ParallelDistrib. Syst., vol. 18, no. 6, pp. 849–862, June 2007.