IJARES/September 2014/Volume-2/Issue-9/1468-1472 **ISSN 2347- 9337**

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN ENGINEERING AND SCIENCE

LESSENING OF NETWORK TRAFFIC TOWARDS OPTIMIZING NETWORK BANDWIDTH

Prem Kumar Naganur[1], N.Ashok[2]

[1]M.Tech Student, Dept of CSE, Chilkur Balaji Institute of Technology, Hyderabad, T.S, India

[2]Associate Professor, Dept of CSE, Chilkur Balaji Institute of Technology, Hyderbad, T.S, India

ABSTRACT:

Cloud computing is a setting for resource sharing devoid of awareness of infrastructure and makes it feasible to access applications and its related data from anywhere at any instant. Modern dispersed file systems in clouds depend on central nodes to administer the metadata information of file systems and to stabilize loads of storage nodes based on metadata. We are concerned in studying the load-rebalancing difficulty in distributed file systems particular for extensive, dynamic as well as data-intensive clouds. Our objective is to assign the chunks of files as consistently as possible between nodes such that no nodes manage an extreme number of chunks. Most existing solutions are intended devoid of considering movement cost as well as node heterogeneity and might set up significant continuation network traffic to distributed hash tables. A novel load-balancing algorithm to handle with the load rebalancing problem in significant, dynamic, as well as dispersed file systems in clouds has been put forward. Our proposal strives to equilibrium the loads of nodes and diminishes demanded movement cost as much as probable, while taking benefit of physical network locality and node heterogeneity. In proposed algorithm, each chunk server node initially estimates whether it is under loaded or overloaded devoid of global knowledge.

Keywords: Distributed hash tables, Load-rebalancing, Cloud computing, File chunks.

1. INTRODUCTION:

Cloud computing is an expertise, where a pool of resources are associated in concealed as well as public networks and to make available these dynamically liable communications in support of application [1]. It is a service oriented and put forward virtualized resources towards cloud users. Distributed file systems are significant building blocks for cloud computing applications on basis of MapReduce programming concept. In such file systems, nodes concurrently serve computing as well as storage functions; a file is partitioned into a number of chunks assigned in separate nodes so that MapReduce tasks are performed in parallel over nodes. Modern dispersed file systems in clouds depend on central nodes to administer the metadata information of file systems and to stabilize loads of storage nodes based on metadata. Even with newest development in dispersed file systems, the central nodes might still be overloaded. As the workload experienced by the name nodes might change over time and no adaptive workload consolidation or migration system is offered to equilibrium the loads among the name nodes, any of name nodes may turn out to be performance bottleneck [2][3]. Load balance between storage nodes is a significant function in clouds. In a load-balanced cloud, resources can be well exploited, maximizing performance of MapReduce-based applications. Recent experience concludes that when number of storage nodes, files as well as number of accesses to files augments linearly, central nodes turn out to be a performance blockage, as they are not capable to put up a huge number of file accesses due to clients as well as MapReduce applications. Consequently depending on central nodes to attempt load imbalance difficulty make worse their heavy loads [4].

2. METHODOLOGY:

Cloud computing make available dynamic provisioning and consequently can distribute machines to store up data and append or eliminate the machines consistent with workload demands. It is a setting for resource sharing devoid of awareness of infrastructure and makes it feasible to access applications and its related data from anywhere at any instant and is on the basis of virtualization technology which is used to distribute data center resources energetically based on demands of application. A novel load-balancing algorithm to handle with the

load rebalancing problem in significant, dynamic, as well as dispersed file systems in clouds has been put forward. Our proposal strives to equilibrium the loads of nodes and diminishes demanded movement cost as much as probable, while taking benefit of physical network locality and node heterogeneity [5]. In lack of representative actual workloads in public domain, we have explored the performance of our proposal as well as compared it against competing algorithms all the way through synthesized probabilistic allocation of file chunks. The synthesis workloads stress test load-balancing algorithms through creating a few storage nodes that are greatly loaded. The simulation consequences are encouraging, indicating that projected algorithm performs extremely well and exhibits a speedy convergence rate. The efficiency and success of our design are additional validated by analytical representation as well as a real functioning with a small-scale cluster setting. In proposed algorithm, each chunk server node initially estimates whether it is under loaded or overloaded devoid of global knowledge [6].

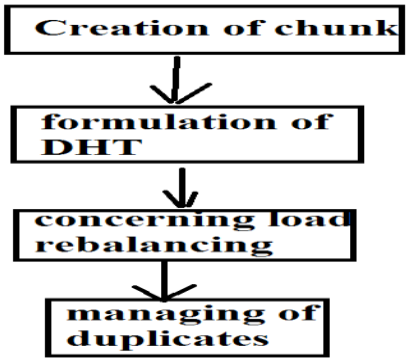

Fig1: An overview of Load Rebalancing Process

3. AN OVERVIEW OF PROPOSED SYSTEM:

We are concerned in studying the load-rebalancing difficulty in distributed file systems particular for extensive, dynamic as well as data-intensive clouds. Such an extensive cloud has numerous nodes. Our objective is to assign the chunks of files as consistently as possible between nodes such that no nodes manage an extreme number of chunks [7]. We intend to decrease network traffic caused by rebalancing loads of nodes as much as likely to make the most of network bandwidth obtainable to usual applications. Moreover, as malfunction is norm, nodes are recently added to maintain general system performance ensuing in heterogeneity of nodes. Exploiting capable nodes to get better system performance is, consequently, demanded. The storage nodes are controlled as a network based on

distributed hash tables discovering a file chunk can merely refer to rapid key search for in distributed hash tables specified that an exceptional handle is assigned towards each file chunk. Distributed hash tables facilitate nodes to self-organize and repair while regularly offer lookup functionality in node dynamism; simplify system provision as well as management. Proposed algorithm functions in a dispersed manner in which nodes carry out their load-balancing tasks autonomously devoid of synchronization or else global knowledge concerning system. Most existing solutions are intended devoid of considering movement cost as well as node heterogeneity and might set up significant continuation network traffic to distributed hash tables. Proposal not only takes benefit of physical network locality in reorganization of file chunks to reduce movement outlay but also make use of capable nodes to get better general system performance. Our algorithm reduces algorithmic transparency introduced to distributed hash tables as much as promising and is assessed all the way through computer simulations [8]. We initially present a load-balancing algorithm, in which every node has global information concerning the system that leads to small movement outlay and quick convergence. The mapping between lightest as well as heaviest nodes at each time in a series can be additional enhanced to reach global load-balanced system state. We have set up our algorithm when each node contains global knowledge of loads of all nodes in system. On the other hand, it is a difficult challenge for every node to contain such comprehensive knowledge in an extensive and active computing environment.

4. CONCLUSION:

Cloud computing is an expertise, where a pool of resources are associated in concealed as well as public networks and to make available these dynamically liable communications in support of application. Distributed file systems are significant building blocks for cloud computing applications on basis of MapReduce programming concept. Recent experience concludes that when number of storage nodes, files as well as number of accesses to files augments linearly, central nodes turn out to be a performance blockage, as they are not capable to put up a huge number of file accesses due to clients as well as MapReduce applications. . In a load-balanced cloud, resources can be well

exploited, maximizing performance of MapReduce-based applications. A novel load-balancing algorithm to handle with the load rebalancing problem in significant, dynamic, as well as dispersed file systems in clouds has been put forward. Our proposal strives to equilibrium the loads of nodes and diminishes demanded movement cost as much as probable, while taking benefit of physical network locality and node heterogeneity. The simulation consequences are encouraging, indicating that projected algorithm performs extremely well and exhibits a speedy convergence rate. The efficiency and success of our design are additional validated by analytical representation as well as a real functioning with a small-scale cluster setting. Distributed hash tables facilitate nodes to self-organize and repair while regularly offer lookup functionality in node dynamism; simplify system provision as well as management. Proposed algorithm functions in a dispersed manner in which nodes carry out their load-balancing tasks autonomously devoid of synchronization or else global knowledge concerning system.

REFERENCES

[1] J.W. Byers, J. Considine, and M. Mitzenmacher, "Simple Load Balancing for Distributed Hash Tables," Proc. First Int'l Workshop Peer-to-Peer Systems (IPTPS '03), pp. 80-87, Feb. 2003.

[2] G.S. Manku, "Balanced Binary Trees for ID Management and Load Balance in Distributed Hash Tables," Proc. 23rd ACM Symp. Principles Distributed Computing (PODC '04), pp. 197-205, July 2004.

[3] A. Bharambe, M. Agrawal, and S. Seshan, "Mercury: Supporting Scalable Multi-Attribute Range Queries," Proc. ACM SIGCOMM '04, pp. 353-366, Aug. 2004.

[4] Y. Zhu and Y. Hu, "Efficient, Proximity-Aware Load Balancing for DHT-Based P2P Systems," IEEE Trans. Parallel and Distributed Systems, vol. 16, no. 4, pp. 349-361, Apr. 2005.

[5] H. Shen and C.-Z. Xu, "Locality-Aware and Churn-Resilient Load Balancing Algorithms in Structured P2P Networks," IEEE Trans. Parallel and Distributed Systems, vol. 18, no. 6, pp. 849-862, June 2007.

[6] Q.H. Vu, B.C. Ooi, M. Rinard, and K.-L. Tan, "Histogram-Based Global Load Balancing in Structured Peer-to-Peer Systems," IEEE Trans. Knowledge Data Eng., vol. 21, no. 4, pp. 595-608, Apr. 2009.

[7] H.-C. Hsiao, H. Liao, S.-S. Chen, and K.-C. Huang, "Load Balance with Imperfect Information in Structured Peer-to-Peer Systems," IEEE Trans. Parallel Distributed Systems, vol. 22, no. 4, pp. 634-649, Apr. 2011.

[8] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Co., 1979.