Abstract

Distributed file systems are the fundamental units for cloud applications where in the data node concurrently serves the computing and storage functions. In these file systems, a file is split by a master node into a set of file chunks and allotted to separate data nodes such that various jobs can be carried out in parallel across the data nodes. However, the unpredictability of the nodes and dynamism in the number of files raise the need for uniform re-distribution of files to prevent the adverse effects of load imbalance.

Hence, the latest enhancement to distributed file systems is a decentralized and asynchronous load rebalancing algorithm that exploits both heterogeneity and movement cost for file chunk allocation among data nodes. But, the load rebalancing protocol has its basis in a randomized method wherein the data node periodically collects and sorts the storage load status of an instance of arbitrary chosen data nodes without considering their computational capabilities or the physical proximity information thereby introducing not only considerable workload on the data nodes but also high overhead on message exchanges among data nodes thus leading to...
reducing scalability. Moreover, the distributed load re-balancing approach does not consider the additional redundant overhead on the data nodes from the federated, load imbalanced master nodes.

In the current study, a completely distributed performance-driven load balancing approach (PDLB) that employs Zero-Hop Hash Table (ZHT) and Modified Firefly Algorithm (MFA) is suggested for coping with the load imbalance issue on both master node and data node. The aim of PDLB is to arrive at data allocations among nodes that could achieve maximum resource utilization at optimized movement cost and minimized message exchanges and algorithmic overhead. The experimental results indicate that PDLB performs better than the earlier distributed protocol about overhead on message exchanges, scalability, movement cost, load imbalance factors as well as algorithmic overheads.

References

1. Grossman, Robert L., Yunhong Gu, Michael Sabala, and Wanzhi Zhang. "Compute and storage clouds using wide area high performance networks." Future Generation Computer Systems 25, no. 2 (2009): 179-183.
2. Xiao, Zhen, Weijia Song, and Qi Chen. "Dynamic resource allocation using virtual machines for cloud computing environment." Parallel and Distributed Systems, IEEE Transactions on 24, no. 6 (2013): 1107-1117.
3. Hsiao, Hung-Chang, Hsueh-Yi Chung, Haiying Shen, and Yu-Chang Chao. "Load rebalancing for distributed file systems in clouds." Parallel and Distributed Systems, IEEE Transactions on 24, no. 5 (2013): 951-962.
4. S. Ghemawat, H. Gobioff, and S.-T. Leung, “The Google File System,” Proc. 19th ACM Symp. Operating Systems Principles (SOSP ’03), pp. 29-43, Oct. 2003.
5. Hadoop Distributed File System, http://hadoop.apache.org/hdfs/, 2012.
6. L. M. Ni and K. Hwang, “Optimal Load Balancing in a Multiple Processor System with Many Job Classes,” IEEE Trans. Software Eng., vol. 11, no. 5, pp. 491–496, May 1985.
7. L. M. Ni, C.-W. Xu, and T. B. Gendreau, “A Distributed Drafting Algorithm for Load Balancing,” IEEE Trans. Software Eng., vol. 11, no. 10, pp. 1153–1161, Oct. 1985.
8. Zhang, Qi, Mohamed Faten Zhani, Shuo Zhang, Quanyan Zhu, Raouf Boutaba, and Joseph L. Hellerstein. "Dynamic energy-aware capacity provisioning for cloud computing environments." In Proceedings of the 9th international conference on Autonomic computing, pp. 145-154. ACM, 2012.
9. Eastlake, Donald, and Paul Jones. "US secure hash algorithm 1 (SHA1)." (2001).
10. Li, Tonglin, Xiaobing Zhou, Kevin Brandstatter, Dongfang Zhao, Ke Wang, Anupam Rajendran, Zhao Zhang, and Ioan Raicu. "ZHT: A light-weight reliable persistent dynamic scalable zero-hop distributed hash table." In Parallel & Distributed Processing (IPDPS), 2013 IEEE 27th International Symposium on, pp. 775-787. IEEE, 2013.
11. McNett, Marvin, Diwaker Gupta, Amin Vahdat, and Geoffrey M. Voelker. "Usher: An Extensible Framework for Managing Clusters of Virtual Machines." In LISA, vol. 7, pp. 1-15. 2007.
12. C. A. Waldspurger, “Memory resource management in VMware ESX server,” in Proc. of the symposium on Operating systems design and implementation (OSDI’02), Aug. 2002.
13. Abu-Libdeh, Hussam, Paolo Costa, Antony Rowstron, Greg O’Shea, and Austin
Donnelly. "Symbiotic routing in future data centers." ACM SIGCOMM Computer Communication Review 41, no. 4 (2011): 51-62.

14. Balasangameshwara, Jasma, and Nedunchezian Raju. "Performance-driven load balancing with a primary-backup approach for computational grids with low communication cost and replication cost." Computers, IEEE Transactions on 62, no. 5 (2013): 990-1003.

15. Naor, Moni, and Udi Wieder. "A simple fault tolerant distributed hash table." In Peer-to-Peer Systems II, pp. 88-97. Springer Berlin Heidelberg, 2003.

16. Li, Tonglin, Xiaobing Zhou, Kevin Brandstatter, Dongfang Zhao, Ke Wang, Anupam Rajendran, Zhao Zhang, and Ioan Raicu. "ZHT: A light-weight reliable persistent dynamic scalable zero-hop distributed hash table." In Parallel & Distributed Processing (IPDPS), 2013 IEEE 27th International Symposium on, pp. 775-787. IEEE, 2013.

17. 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.

18. S. Surana, B. Godfrey, K. Lakshminarayanan, R. Karp, and I. Stoica, “Load Balancing in Dynamic Structured P2P Systems,” Performance Evaluation, vol. 63, no. 6, pp. 217-240, Mar. 2006.

19. Birman, Ken. "The promise, and limitations, of gossip protocols." ACM SIGOPS Operating Systems Review 41, no. 5 (2007): 8-13.

20. I. Stoica, R. Morris, D. Liben-Nowell, D.R. Karger, M.F. Kaashoek, F. Dabek, and H. Balakrishnan, “Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications,” IEEE/ACM Trans. Networking, vol. 11, no. 1, pp. 17-21, Feb. 2003.

21. A. Rowstron and P. Druschel, “Pastry: Scalable, Distributed Object Location and Routing for Large-Scale Peer-to-Peer Systems,” Proc. IFIP/ACM Int’l Conf. Distributed Systems Platforms Heidelberg, pp. 161-172, Nov. 2001.

22. G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels, “Dynamo: Amazon’s Highly Available Key-Value Store,” Proc. 21st ACM Symp. Operating Systems Principles (SOSP ’07), pp. 205-220, Oct. 2007.

23. HDFS Federation, http://hadoop.apache.org/common/docs/r0.23.0/hadoop-yarn/hadoop-yarn-site/Federation.html

24. Sagan, Hans. Space-filling curves. Vol. 18. New York: Springer-Verlag, 1994.

25. Wolski, Rich, Neil T. Spring, and Jim Hayes. "The network weather service: a distributed resource performance forecasting service for metacomputing." Future Generation Computer Systems 15, no. 5 (1999): 757-768.

26. Heger, Dominique. "Hadoop Performance Tuning-A Pragmatic & Iterative Approach." CMG Journal (2013).

27. Balasangameshwara, Jasma, and Nedunchezian Raju. "A hybrid policy for fault tolerant load balancing in grid computing environments." Journal of Network and Computer Applications 35, no. 1 (2012): 412-422.

28. Wang, Ke, Abhishek Kulkarni, Michael Lang, Dorian Arnold, and Ioan Raicu. "Using simulation to explore distributed key-value stores for extreme-scale system services." In Proceedings of SC13: International Conference for High Performance Computing, Networking, Storage and Analysis, p. 9. ACM, 2013.

29. Raicu, Ioan, Ian T. Foster, and Pete Beckman. "Making a case for distributed file systems at exascale." In Proceedings of the third international workshop on Large-scale system
and application performance, pp. 11-18. ACM, 2011.

30. Zhu, Yingwu, and Yiming Hu. "Efficient, proximity-aware load balancing for DHT-based P2P systems." Parallel and Distributed Systems, IEEE Transactions on 16, no. 4 (2005): 349-361.

31. Shen, Haiying, and Cheng-Zhong Xu. "Locality-aware and churn-resilient load-balancing algorithms in structured peer-to-peer networks." Parallel and Distributed Systems, IEEE Transactions on 18, no. 6 (2007): 849-862.

32. Hsiao, Hung-Chang, Hao Liao, Ssu-Ta Chen, and Kuo-Chan Huang. "Load balance with imperfect information in structured peer-to-peer systems." Parallel and Distributed Systems, IEEE Transactions on 22, no. 4 (2011): 634-649.

33. Hua, Xiayu, Hao Wu, Zheng Li, and Shangping Ren. "Enhancing throughput of the Hadoop Distributed File System for interaction-intensive tasks." Journal of Parallel and Distributed Computing 74, no. 8 (2014): 2770-2779.

34. Yang, Xin-She. Nature-inspired metaheuristic algorithms. Luniver press, 2010.

35. Tilahun, Surafel Luleseged, and Hong Choon Ong. "Modified firefly algorithm." Journal of Applied Mathematics 2012 (2012).

36. Chatterjee, A.; Mahanti, G. K.; Chatterjee, A. (2012). "Design of a fully digital controlled reconfigurable switched beam conconcentric ring array antenna using firefly and particle swarm optimization algorithm". Progress in Electromagnetic Research B 36: 113–131.

Index Terms

Computer Science Distributed Systems

Keywords

Map Reduce, Hadoop Distributed File System, Load Balancing, Zero-Hop Hash Table, Firefly Algorithm