References

[1] InternetLiveStats, "Google Search Statistics," 2020, https://www.internetlivestats.com/google-search-statistics/#:~:text=Google%20now%20processes%20over%2040%2C000,trillion%20searches%20per%20year%20worldwide., Last accessed on 2020-03-15.

[2] Oberlo, "10 Facebook Stats Every Marketer Should Know in 2020 [Infographic]," 2020, https://www.oberlo.com/blog/facebook-statistics, Last accessed on 2020-03-15.

[3] C. Kopparapu, *Load balancing servers, firewalls, and caches*. John Wiley & Sons, 2002.

[4] N. J. Kansal and I. Chana, "Cloud load balancing techniques: A step towards green computing," *IJCSI International Journal of Computer Science Issues*, vol. 9, no. 1, pp. 238–246, 2012.

[5] "Load balancing cloud computing: state of art, author=Khiyaita, A and El Bakkali, H and Zbakh, M and El Kettani, Dafir," in *2012 National Days of Network Security and Systems*. IEEE, 2012, pp. 106–109.

[6] S. Kaur, K. Kumar, J. Singh, and N. S. Ghumman, "Round-robin based load balancing in Software Defined Networking," in *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*. IEEE, 2015, pp. 2136–2139.

[7] R. Wang, D. Butnariu, J. Rexford *et al.*, "OpenFlow-Based Server Load Balancing Gone Wild." *Hot-ICE*, vol. 11, pp. 12–12, 2011.

[8] P. Patel, D. Bansal, L. Yuan, A. Murthy, A. Greenberg, D. A. Maltz, R. Kern, H. Kumar, M. Zikos, H. Wu *et al.*, "Ananta: Cloud scale load balancing," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 207–218, 2013.

[9] F5 Networks, "HTTP Load Balancing," 2020, https://docs.nginx.com/nginx/admin-guide/load-balancer/http-load-balancer/, Last accessed on 2020-03-15.

[10] V. Sreenivas, M. Prathap, and M. Kemal, "Load balancing techniques: Major challenge in Cloud Computing-a systematic review," in *2014 International Conference on Electronics and Communication Systems (ICECS)*. IEEE, 2014, pp. 1–6.

[11] The Apache Software Foundation, "Welcome to Apache Axis2/Java," 2018, http://axis.apache.org/axis2/java/core/, Last accessed on 2020-03-15.

[12] HAProxy, "HAProxy," 2020, http://www.haproxy.org/, Last accessed on 2020-03-15.

[13] Virtual Graffiti Inc, "Fortinet Coyote Point Equalizer ADCs," 2020, https://www.avfirewalls.com/Equalizer-ADCs.asp, Last accessed on 2020-03-15.

[14] P. M. Shameem and R. Shaji, "A methodological survey on load balancing techniques in cloud computing," *International Journal of Engineering and Technology (IJET)*, vol. 4, no. 5, pp. 3801–3812, 2013.

[15] S. Rajoriya *et al.*, "Load balancing techniques in cloud computing: an overview," *International Journal of Science and Research*, vol. 3, 2014.

[16] R. P. Padhy and P. Rao, "Load balancing in cloud computing systems," Ph.D. dissertation, 2011.

[17] Z. Zhang and X. Zhang, "A load balancing mechanism based on ant colony and complex network theory in open cloud computing federation," in *2010 The 2nd International Conference on Industrial Mechatronics and Automation*, vol. 2. IEEE, 2010, pp. 240–243.

[18] M. Randles, D. Lamb, and A. Taleb-Bendiab, "A comparative study into distributed load balancing algorithms

for cloud computing," in *2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops*. IEEE, 2010, pp. 551–556.

[19] H. Mehta, P. Kanungo, and M. Chandwani, "Decentralized content aware load balancing algorithm for distributed computing environments," in *Proceedings of the International Conference & Workshop on Emerging Trends in Technology*, 2011, pp. 370–375.

[20] A. M. Nakai, E. Madeira, and L. E. Buzato, "Load balancing for internet distributed services using limited redirection rates," in *2011 5th Latin-American Symposium on Dependable Computing*. IEEE, 2011, pp. 156–165.

[21] Y. Lu, Q. Xie, G. Kliot, A. Geller, J. R. Larus, and A. Greenberg, "Join-Idle-Queue: A novel load balancing algorithm for dynamically scalable web services," *Performance Evaluation*, vol. 68, no. 11, pp. 1056–1071, 2011.

[22] X. Liu, L. Pan, C.-J. Wang, and J.-Y. Xie, "A lock-free solution for load balancing in multi-core environment," in *2011 3rd International Workshop on Intelligent Systems and Applications*. IEEE, 2011, pp. 1–4.

[23] V. Nae, R. Prodan, and T. Fahringer, "Cost-efficient hosting and load balancing of massively multiplayer online games," in *2010 11th IEEE/ACM International Conference on Grid Computing*. IEEE, 2010, pp. 9–16.

[24] D. Pariag, T. Brecht, A. Harji, P. Buhr, A. Shukla, and D. R. Cheriton, "Comparing the performance of web server architectures," *ACM SIGOPS Operating Systems Review*, vol. 41, no. 3, pp. 231–243, 2007.

[25] A. S. Harji, P. A. Buhr, and T. Brecht, "Comparing high-performance multi-core web-server architectures," in *Proceedings of the 5th Annual International Systems and Storage Conference*, 2012, pp. 1–12.

[26] M. Thompson, D. Farley, M. Barker, P. Gee, and A. Stewart, "Disruptor: High performance alternative to bounded queues for," 2011.

[27] B. Erb, "Concurrent Programming for Scalable Web Architectures," Diploma Thesis, Institute of Distributed Systems, Ulm University, April 2012. [Online]. Available: http://www.benjamin-erb.de/thesis

[28] G. Banga, J. C. Mogul, P. Druschel *et al.*, "A Scalable and Explicit Event Delivery Mechanism for UNIX." in *USENIX Annual Technical Conference, General Track*, 1999, pp. 253–265.

[29] W. R. Stevens and T. Narten, "UNIX network programming," *ACM SIGCOMM Computer Communication Review*, vol. 20, no. 2, pp. 8–9, 1990.

[30] S. Vinoski, "Concurrency with erlang," *IEEE Internet Computing*, vol. 11, no. 5, pp. 90–93, 2007.

[31] J. Ousterhout, "Why threads are a bad idea (for most purposes)," in *Presentation given at the 1996 Usenix Annual Technical Conference*, vol. 5. San Diego, CA, USA, 1996.

[32] Welsh, Matt and Culler, David and Brewer, Eric, "SEDA: an architecture for well-conditioned, scalable internet services," *ACM SIGOPS Operating Systems Review*, vol. 35, no. 5, pp. 230–243, 2001.

[33] "Flash: An efficient and portable Web server., author=Pai, Vivek S and Druschel, Peter and Zwaenepoel, Willy," in *USENIX Annual Technical Conference, General Track*, 1999, pp. 199–212.

[34] J. C. Hu, I. Pyarali, and D. C. Schmidt, "Measuring the impact of event dispatching and concurrency models on web server performance over high-speed networks," in *GLOBECOM 97. IEEE Global Telecommunications Conference. Conference Record*, vol. 3. IEEE, 1997, pp. 1924–1931.

[35] The Apache Software Foundation, "Apache HTTP Server Project," 2020, https://httpd.apache.org/, Last accessed on 2020-03-15.

[36] Microsoft, “Microsoft IIS,” 2020, https://www.iis.net/, Last accessed on 2020-03-15.

[37] IBM, “IBM WebSphere Application Server,” 2020, https://www.ibm.com/cloud/websphere-application-platform/, Last accessed on 2020-03-15.

[38] D. C. Schmidt, “Reactor: An object behavioral pattern for concurrent event demultiplexing and dispatching,” 1995.

[39] I. Pyarali, T. Harrison, D. C. Schmidt, and T. D. Jordan, “Proactor-an object behavioral pattern for demultiplexing and dispatching handlers for asynchronous events,” 1997.

[40] N. Zeldovich, A. Yip, F. Dabek, R. T. Morris, D. Mazieres, and M. F. Kaashoek, “Multiprocessor Support for Event-Driven Programs.” in *USENIX Annual Technical Conference, General Track*, 2003, pp. 239–252.

[41] J. R. Von Behren, J. Condit, and E. A. Brewer, “Why Events Are a Bad Idea (for High-Concurrency Servers).” in *HotOS*, 2003, pp. 19–24.

[42] Q. Fan and Q. Wang, “Performance comparison of web servers with different architectures: A case study using high concurrency workload,” in *2015 Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*. IEEE, 2015, pp. 37–42.

[43] E. Al-Rayis and H. Kurdi, “Performance Analysis of load balancing Architectures in Cloud computing,” in *2013 European Modelling Symposium*. IEEE, 2013, pp. 520–524.

[44] G. S. Choi, J.-H. Kim, D. Ersoz, and C. R. Das, “A multi-threaded PIPELINED Web server architecture for SMP/SoC machines,” in *Proceedings of the 14th international conference on World Wide Web*, 2005, pp. 730–739.

[45] G. Upadhyaya, V. S. Pai, and S. P. Midkiff, “Expressing and exploiting concurrency in networked applications with aspen,” in *Proceedings of the 12th ACM SIGPLAN symposium on Principles and practice of parallel programming*, 2007, pp. 13–23.

[46] B. Burns, K. Grimaldi, A. Kostadinov, E. D. Berger, and M. D. Corner, “Flux: A language for programming high-performance servers,” *Computer Science Department Faculty Publication Series*, p. 53, 2006.

[47] I. Voras and M. Zagar, “Characteristics of multithreading models for high-performance io driven network applications,” in *AFRICON 2009*. IEEE, 2009, pp. 1–6.

[48] OW2, “RUBBoS,” 2020, https://projects.ow2.org/view/rubbos/, Last accessed on 2020-03-15.