10. REFERENCES

[1] S. Almeida, J. Leitão, and L. Rodrigues. Chain- reaction: A causal+ consistent datastore based on chain replication. In *Proc. of 8th European Conference on Computer Systems*, EuroSys'13, pages 85–98, 2013.

[2] R. Ananthanarayanan, V. Basker, S. Das, A. Gupta, H. Jiang, T. Qiu, A. Reznichenko, D. Ryabkov, M. Singh, and S. Venkataraman. Photon: Fault- tolerant and scalable joining of continuous data streams. In *SIGMOD '13: Proc. of 2013 international conf. on Management of data*, pages 577–588, 2013.

[3] J. Baker, C. Bond, J. C. Corbett, J. Furman, A. Khorlin, J. Larson, J.-M. Leon, Y. Li, A. Lloyd, and V. Yushprakh. Megastore: Providing scalable, highly available storage for interactive services. In *Proc. of the Conference on Innovative Data system Research (CIDR)*, pages 223–234, 2011.

[4] P. Bernstein and N. Goodman. Concurrency control in distributed database systems. *ACM Computing Surveys*, 13(2), January 1981.

[5] N. Bronson et al. Tao: Facebook's distributed data store for the social graph. In *Proc. of the 2013 USENIX Annual Technical Conference*, pages 49–60, 2013.

[6] D. G. Campbell, G. Kakivaya, and N. Ellis. In *Proc. of the 2010 ACM SIGMOD International Conference on Management of Data*, pages 1021–1024.

[7] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber. Bigtable: A distributed storage system for structured data. *ACM Trans. Comput. Syst.*, 26(2):4:1–4:26, June 2008.

[8] B. F. Cooper, A. Silberstein, E. Tam, R. Ramakrishnan, and R. Sears. Benchmarking cloud serving systems with ycsb. In *Proc. of the 1st ACM Symposium on Cloud Computing*, pages 143–154, 2010.

[9] J. C. Corbett et al. Spanner: Google's globally-distributed database. In *Proc. of the 10th USENIX Conference on Operating Systems Design and Implementation*, OSDI'12, pages 251–264, 2012.

[10] G. DeCandia et al. In *Proc. of the 21st ACM Symposium on Operating Systems Principles*, pages 205–220.

[11] A. K. Elmagarmid. A survey of distributed deadlock detection algorithms. *SIGMOD Rec.*, 15(3):37–45, Sept. 1986.

[12] S. Elnikety, W. Zwaenepoel, and F. Pedone. Database replication using generalized snapshot isolation. In *Proceedings of the 24th IEEE Symposium on Reliable Distributed Systems*, SRDS '05, pages 73–84, Washington, DC, USA, 2005. IEEE Computer Society.

[13] L. Glendenning, I. Beschastnikh, A. Krishnamurthy, and T. Anderson. Scalable consistency in Scatter. In *Proc. of the 23rd ACM Symposium on Operating Systems Principles*, SOSP '11, pages 15–28, 2011.

[14] J. Gray and L. Lamport. Consensus on transaction commit. *ACM Trans. Database Syst.*, 31(1):133–160, Mar. 2006.

[15] T. Hoff. Latency is everywhere and it costs you sales - how to crush it. Post at the High Scalability blog. http://tinyurl.com/5g8mp2, 2009.

[16] T. Kraska, G. Pang, M. J. Franklin, S. Madden, and A. Fekete. Mdcc: Multi-data center consistency. In *Proc. of the 8th ACM European Conference on Computer Systems*, EuroSys '13, pages 113–126, 2013.

[17] A. Lakshman and P. Malik. Cassandra: A decentralized structured storage system. *SIGOPS Oper. Syst. Rev.*, 44(2):35–40, Apr. 2010.

[18] L. Lamport. Time, clocks, and the ordering of events in a distributed system. *Commun. ACM*, 21(7):558–565, July 1978.

[19] L. Lamport. The part-time parliament. *ACM Trans. Comput. Syst.*, 16(2):133–169, May 1998.

[20] L. Lamport, D. Malkhi, and L. Zhou. Reconfiguring a state machine. *ACM SIGACT News*, 41(1):63–73, Mar. 2010.

[21] W. Lloyd, M. J. Freedman, M. Kaminsky, and D. G. Andersen. In *Proc. of the Twenty-Third ACM Symposium on Operating Systems Principles*, pages 401–416.

[22] W. Lloyd, M. J. Freedman, M. Kaminsky, and D. G. Andersen. Stronger semantics for low-latency geo-replicated storage. In *Proc. of the 10th USENIX Conference on Networked Systems Design and Implementation*, NSDI'13, pages 313–328, 2013.

[23] H. Mahmoud, F. Nawab, A. Pucher, D. Agrawal, and A. El Abbadi. Low-latency multi-datacenter databases using replicated commit. *Proc. VLDB Endow.*, 6(9):661–672, July 2013.

[24] M. Saeida Ardekani, P. Sutra, and M. Shapiro. Non-Monotonic Snapshot Isolation: scalable and strong consistency for geo-replicated transactional systems. In *Proc. of the 32nd IEEE Symposium on Reliable Distributed Systems (SRDS 2013)*, pages 163–172, 2013.

[25] M. Saeida Ardekani, P. Sutra, M. Shapiro, and N. Preguiça. On the scalability of snapshot isolation. In *Euro-Par 2013 Parallel Processing*, volume 8097 of *LNCS*, pages 369–381. Springer, 2013.

[26] F. B. Schneider. Implementing fault-tolerant services using the state machine approach: A tutorial. *ACM Comput. Surv.*, 22(4):299–319, Dec. 1990.

[27] J. Shute, R. Vingralek, B. Samwel, B. Handy, C. Whipkey, E. Rollins, M. Oancea, K. Littlefield, D. Menestrina, S. Ellner, J. Cieslewicz, I. Rae, T. Stancescu, and H. Apte. F1: A distributed sql database that scales. *Proc. VLDB Endow.*, 6(11):1068–1079, Aug. 2013.

[28] Y. Sovran, R. Power, M. K. Aguilera, and J. Li. Transactional storage for geo-replicated systems. In *Proc. of the 23rd ACM Symposium on Operating Systems Principles*, SOSP '11, pages 385–400, 2011.

[29] I. Zhang, N. K. Sharma, A. Szekeres, A. Krishnamurthy, and D. R. K. Ports. Building consistent transactions with inconsistent replication. In *Proc. of the 25th ACM Symposium on Operating Systems Principles (SOSP)*, pages 263–278, 2015.

[30] Y. Zhang, R. Power, S. Zhou, Y. Sovran, M. Aguilera, and J. Li. Transaction chains: Achieving serializability with low latency in geo-distributed storage systems. In *Proc. of the 24th ACM Symposium on Operating Systems Principles*, SOSP, pages 276–291, 2013.