Stream Processing Systems Have Arrived at the Big Data Party. But Where Are All the Benchmarks?

Zubair Nabi
IBM Research – Ireland

ABSTRACT

Stream processing systems have now become an integral part of the Big Data ecosystem. Unfortunately, streaming benchmarks have not followed suit leading to non-representative benchmarking of systems. Benchmarks in general have many use cases including: (a) comparing two or more systems, (b) matching applications and workloads to systems, and (c) configuring and optimizing a system. Due to these factors, the Big Data domain has many benchmarks including TPC for DBMS, workload traces for MapReduce, and synthetic workload generators for NoSQL stores and filesystems. In contrast, the stream processing community only has a single benchmark: Linear Road [2]. Even this benchmark is outdated now as at the time of its development, stream processing was numerical-data centric due to the focus on sensor networks and financial transactions while now a large class of Web 2.0 applications requires text analysis. As a result, almost all recent scholarly work has relied on non-standard benchmarks [1, 3].

To remedy this, stream processing benchmarks need to be designed by taking production applications and workloads into consideration. These benchmark suites need to also be mindful of the idiosyncrasies of streaming applications, such as tuple data type and cardinality, and data rate and distribution while allowing different characteristics of the target systems to be tested, such as performance, scalability, and fault-tolerance. To enable this, real-world Web 2.0 production environments need to share their workload traces with the community and academia needs to think benchmarks first. Only then can stream processing systems be properly compared, contrasted, and enhanced.

BODY

The ubiquity of stream processing systems and applications necessitates the development of a real-world TPC-like benchmark suite.

REFERENCES

[1] T. Akidau, A. Balikov, K. Bekiroğlu, S. Chernyak, J. Haberman, R. Lax, S. McVeety, D. Mills, P. Nordstrom, and S. Whittle. MillWheel: Fault-tolerant Stream Processing at Internet Scale. *Proc. VLDB Endow.*, 6(11):1033–1044, Aug. 2013.

[2] A. Arasu, M. Cherniack, E. Galvez, D. Maier, A. S. Maskey, E. Ryvkina, M. Stonebraker, and R. Tibbetts. Linear Road: A Stream Data Management Benchmark. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30*, VLDB '04, pages 480–491. VLDB Endowment, 2004.

[3] M. Zaharia, T. Das, H. Li, T. Hunter, S. Shenker, and I. Stoica. Discretized Streams: Fault-tolerant Streaming Computation at Scale. In *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*, SOSP '13. ACM, 2013.

Volume 3 of Tiny Transactions on Computer Science