

BENCHMARKING GEO-DISTRIBUTED DATABASES

Evaluating Performance using the Product-Parts-Supplier Workload



Eduard-Alex Mihai (emihai@tudelft.nl) | Responsible Professor: Asterios Katsifodimos | Supervisor: Oto Mráz

1. INTRODUCTION

Geo-Distributed Databases: replicate data across regions.

- Crucial for **low-latency, fault-tolerant** services.
- Challenging to coordinate **cross-region transactions**.

Why a New Benchmark?

Existing workloads (TPC-C [1] and YCSB+T [2]) are limited.

RESEARCH QUESTION

*Can a richer workload **uncover design trade-offs** that standard suites miss in geo-distributed settings?*

Method: Building a benchmarking framework based on the PPS workload, and evaluate four representative systems.

2. BACKGROUND AND IMPLEMENTATION

Evaluated Database Systems:

- **Calvin** orders transactions via a global sequencer [3].
- **SLOG** orders only the multi-home transactions [4].
- **Detock** is based on a dependency graph resolution [5].
- **Janus** unifies consensus and concurrency control [6].

PPS Workload: simulates a supply chain management system involving products, parts, and suppliers (Fig. 1). It supports configurable transactions (Fig. 2).

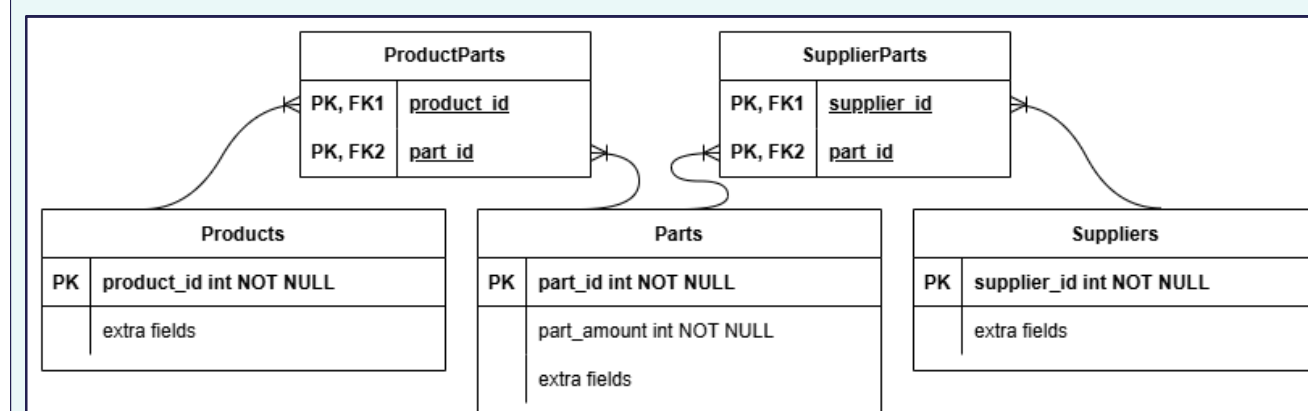


Figure 1: PPS Entity Relationship Diagram

Transaction Type	Ratio
OrderProduct	80%
GetPartsByProduct	8%
UpdateProductPart	8%
GetPart	2%
GetProduct	2%

Figure 2: Transactions Types and Used Ratios

3. EXPERIMENTAL SETUP AND RESULTS

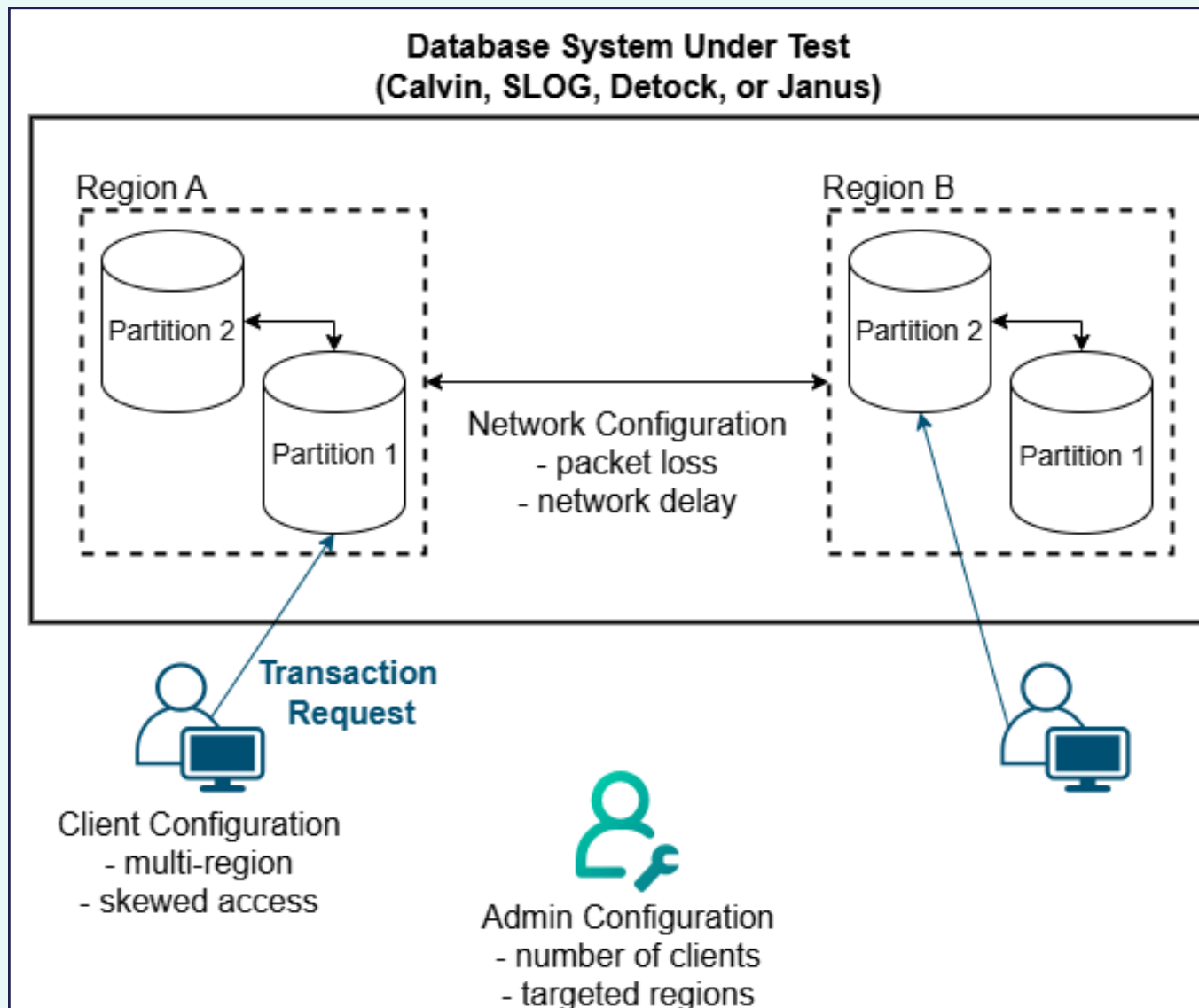


Figure 3: Architecture of the Benchmarking Framework

Benchmark Implementation: Our benchmarking framework (Fig. 3) allows precise control over several configurations:

- **Client:** data access pattern and contention level.
- **Admin:** number and placement of the clients.
- **Network:** latency and link reliability.

4. DISCUSSION

Advantages of PPS:

- **Dependent Transactions:** abort rates in case of conflicts.
 - Visible in all systems in the skew scenario (Fig. 6 & Fig. 7).
- **Longer Transaction Footprint:** effects of larger accessed sets.
 - Visible for Janus in baseline and scalability (Fig. 4 & Fig. 8).
- **Control over Multi-Home:** generation of hotspots and regional bias.
 - Visible for Detock and SLOG in sunflower (Fig. 5).

Limitations of PPS:

- **Limited Writes:** weak write access patterns.
- **Limited Multi-Home:** only *OrderProduct* can be multi-home.

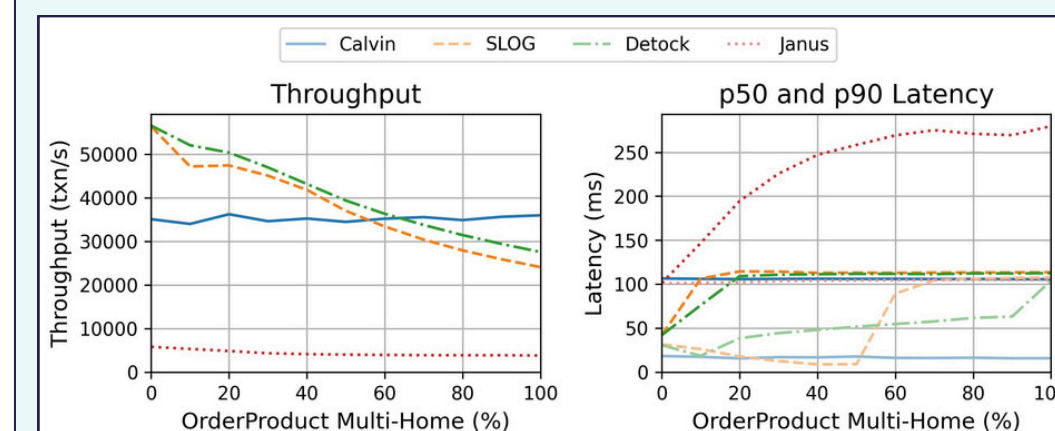


Figure 4: Throughput and Latency for Baseline Scenario

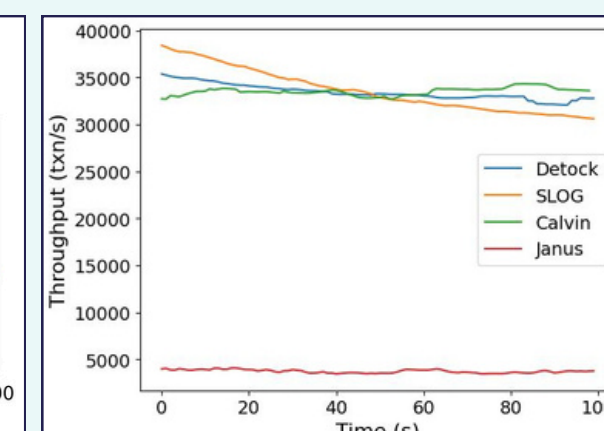


Figure 5: Throughput for Sunflower Scenario

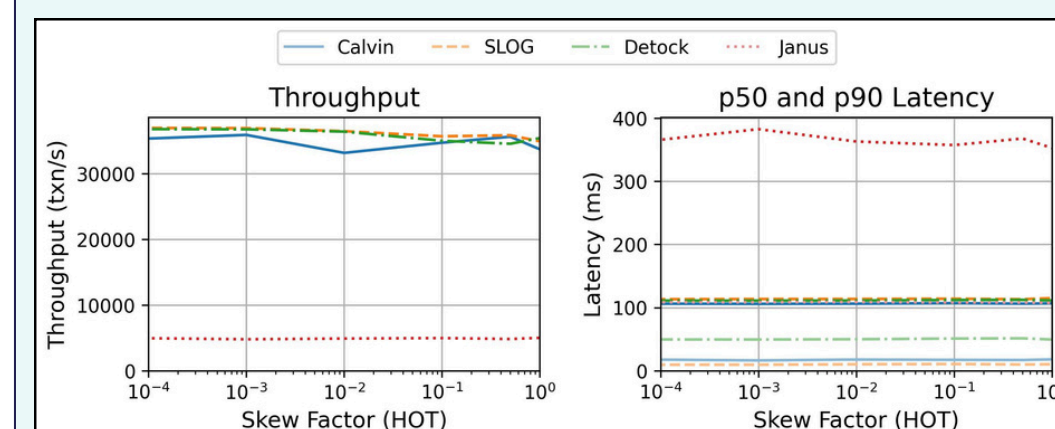


Figure 6: Throughput and Latency for Skew Scenario

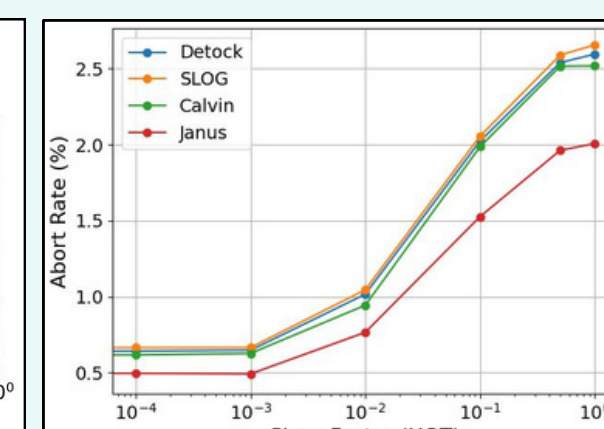


Figure 7: Abort Rate for Skew Scenario

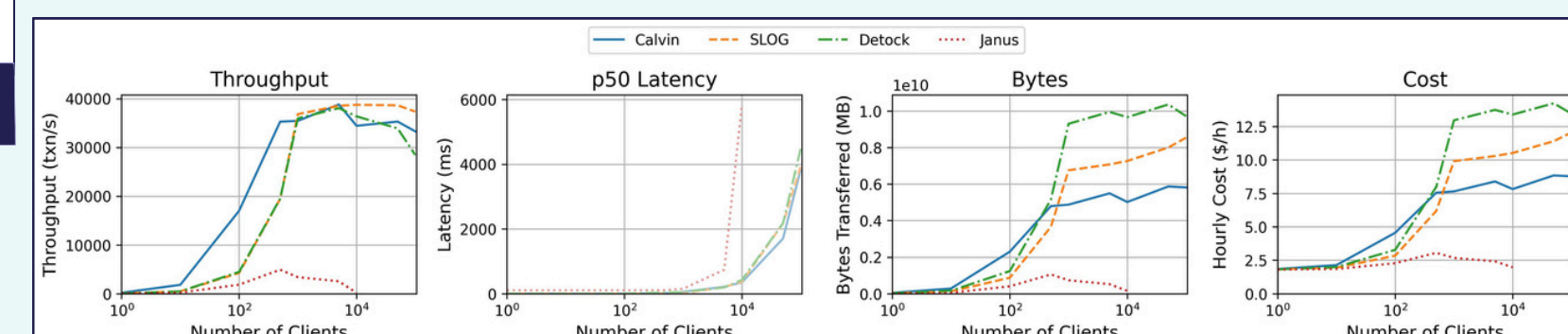


Figure 8: All Collected Metrics for Scalability Scenario

5. CONCLUSION

The benchmark fills **key evaluation gaps** left by the standard workload, but PPS has **several limitations**.

Future work could:

- Add **write-heavy & cross-region transactions**.
- Explore **server-side dependent transactions**.

REFERENCES

- [1] S. T. Leutenegger and D. Dias, "A modeling study of the tpc-c benchmark," ACM Sigmod Record, vol. 22, no. 2, pp. 22–31, 1993.
- [2] A. Dey, A. Fekete, R. Nambiar, and U. R'ohm, "Ycsb+ t: Benchmarking web-scale transactional databases," in 2014 IEEE 30th International Conference on Data Engineering Workshops. IEEE, 2014, pp. 223–230.
- [3] Alexander Thomson, Thaddeus Diamond, Shu-Chun Weng, Kun Ren, Philip Shao, and Daniel J Abadi. Calvin: fast distributed transactions for partitioned database systems. In Proceedings of the 2012 ACM SIGMOD international conference on management of data, pages 1–12, 2012.
- [4] Kun Ren, Dennis Li, and Daniel J Abadi. Slog: Serializable, low-latency, geo-replicated transactions. Proceedings of the VLDB Endowment, 12(11), 2019.
- [5] Cuong DT Nguyen, Johann K Miller, and Daniel J Abadi. Detock: High performance multi-region transactions at scale. Proceedings of the ACM on Management of Data, 1(2):1–27, 2023.
- [6] S. Mu, L. Nelson, W. Lloyd, and J. Li, "Consolidating concurrency control and consensus for commits under conflicts," in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 2016, pp. 517–532.