UNIVERSITY OF WATERLOO
FACULTY OF MATHEMATICS
David R. Cheriton School
of Computer Science

NetStore: Leveraging Network Optimizations to Improve Distributed Transaction Processing Performance

Xu Cui, Michael Mior, Supervisors: Bernard Wong, Khuzaima Daudjee
David R. Cheriton School of Computer Science

Background

Distributed Storage Systems in the Cloud

- The volume of data generated has increased exponentially in the past decade.
- Distributed storage systems are needed to store data across multiple servers.
- The network can become a performance bottleneck.
- Cloud tenants cannot control the cloud network.
- Application level optimizations rely only on static information (i.e. fetch data from a nearby server)

A Datacenter Network

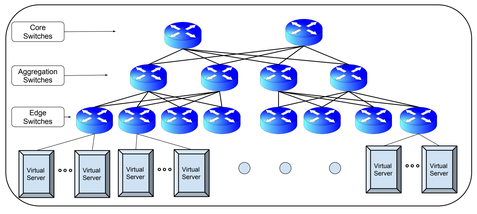

Figure 1: Multi-Rooted Tree Topology

- Multiple unique paths between pairs of servers.
- Part of the network may become congested for a period of time.

NetStore

A network-aware transaction processing system which applies three optimization techniques across network layer and database layer to improve performance.

Least Bottlenecked Path (LBP)

- Network-aware path selection.

Opportunistic Load Redistribution(OLR)

- Designed to redistribute network load.

Earliest Expected Job First (EEJF)

- Designed to further reduce the load on possibly congested links.

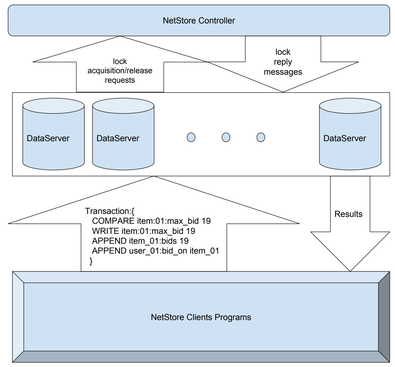

Figure 2: NetStore Architecture

Opportunistic Load Redistribution

OLR is a database layer optimization that effectively reduces the load on network links.

- OLR takes the advantage of read operation results by creating temporary replicas.
- Avoids complex cache eviction implementations.
- Avoids communication costs for cache invalidation.

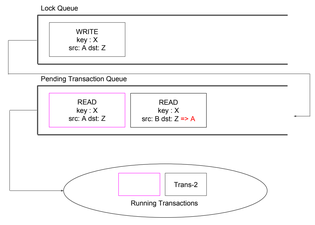

Figure 3: OLR Example Part I

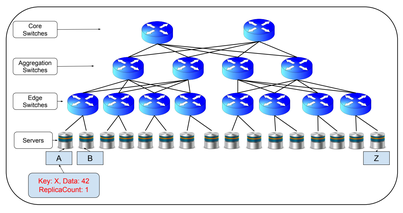

Figure 4: OLR Example Part II

Least Bottlenecked Path

LBP offers network-aware path selection.

- The NetStore controller has global view of the network.
- **LBP** exploits dynamic flow count information on each link to compute bandwidth allocation for each new flow.
- **LBP** makes informed routing decisions based on dynamic flow information.

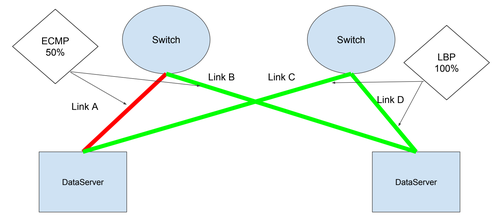

Figure 5: LBP Example

Earliest Expected Job First

EEJF is designed to delay the new flows that may traverse congested links.

- **EEJF** utilizes a network-aware performance model of the underlying system.
- **EEJF** uses this model to predict the expected completion time of a transaction.
- Orders transactions using this expected completion time.

Performance Evaluation

- Mininet is used in a 9-node cluster to emulate a multi-rooted tree topology with 1Gbps links.
- There are a total of eight Top-of-Rack switches and each is connected to eight virtual servers.
- A modified version of the RUBiS benchmark is utilized to evaluate NetStore.
- Equal-Cost Multi-Path (ECMP) routing is used as a baseline for comparison.

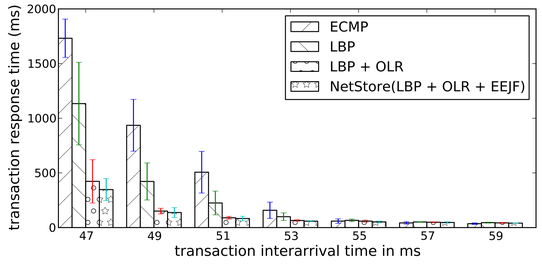

Figure 6: ECMP vs NetStore - Average transaction completion time.

- NetStore has reduced the average transaction completion time by 85% at an interarrival rate of 49 milliseconds.
- NetStore is consistently performing over 50% better than ECMP while the network is congested.
- This performance improvement is achieved without sacrificing the system throughput.

Conclusion

- NetStore bridges the gap between network research and distributed database research to avoid transaction performance deterioration due to network saturation.
- NetStore employs cross-layer optimizations that rely on dynamic network information to improve transaction performance.