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 we generate has increased exponentially in the past decade.
- We need distributed storage systems to store data across multiple servers.
- This makes the network a potential 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

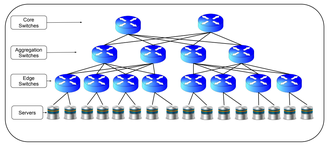

Figure 1: System Overview

- 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.

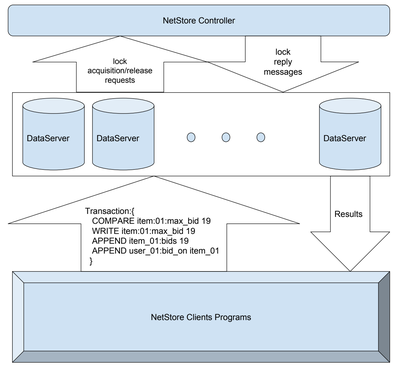

Figure 2: NetStore Architecture

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.

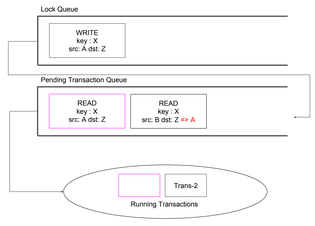

Figure 3: An OLR Example Part I

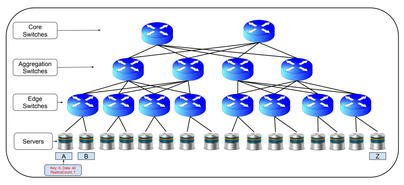

Figure 4: An OLR Example Part II

LEAST BOTTLENECKED PATH

LBP offers network-aware path selection.

- The NetStore controller has the global view of the network.
- We use 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.

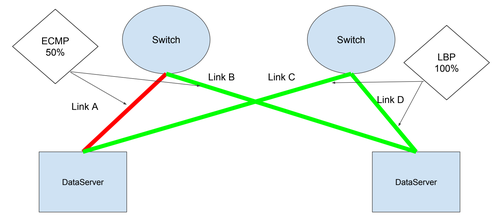

Figure 5: A LBP Example

EARLIEST EXPECTED JOB FIRST

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

- We built a network-aware performance model of the underlying system.
- Uses this model to predict the expected completion time of a transaction.
- Orders the transactions using this expected completion time.

PERFORMANCE EVALUATION

- We use Mininet in a 9-node cluster to emulate a multi-rooted tree topology with 1Gbps links and with a total of 64 virtual servers.
- We use a modified version of the RUBiS benchmark to evaluate NetStore against the Equal-Cost Multi-Path (ECMP) routing algorithm.

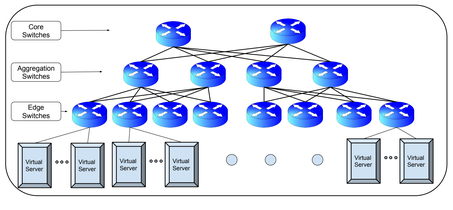

Figure 6: Testbed Network Topology Setup

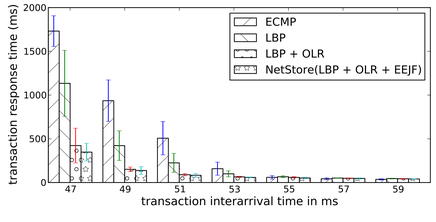

Figure 7: ECMP vs NetStore - Average transaction completion time.

- 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 without sacrificing throughput while the network is saturated.

CONCLUSION

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