DeTail: Reducing the Flow Completion Time Tail in Datacenter Networks

David Zats, Tathagata Das, Prashanth Mohan,
Dhruba Borthakur, Randy Katz

Presented by Alexander Pokluda

February 6, 2013

The Problem

Sophisticated Web Applications

- Rendering a page may require hundreds of requests to back-end servers
- Strict page rendering deadlines of 200-300ms must be met to ensure a positive user experience

Network Complications

- Typically a few responses arrive late giving us *long tailed flow completion times*
- Web applications must choose between sacrificing either quality or responsiveness
- Either option leads to financial loss

=

Network Performance Factors

- Application workflows depend on performance of underlying network flows
- Congestion can cause round-trip-times (RTTs) to form a long-tailed distribution
- Congestion leads to
  - Packet loss and retransmissions
  - Uneven load balancing
  - Priority inversion
- Each contributes to increasing long tail of flow completion, especially for latency-sensitive short flows critical for page creation

Reducing the Flow Completion Time Tail

- Flash congestion can be reduced if it can be detected early enough
- DeTail addresses this challenge by constructing a cross-layer network stack that detects congestion at lower layers to drive upper layer routing decisions

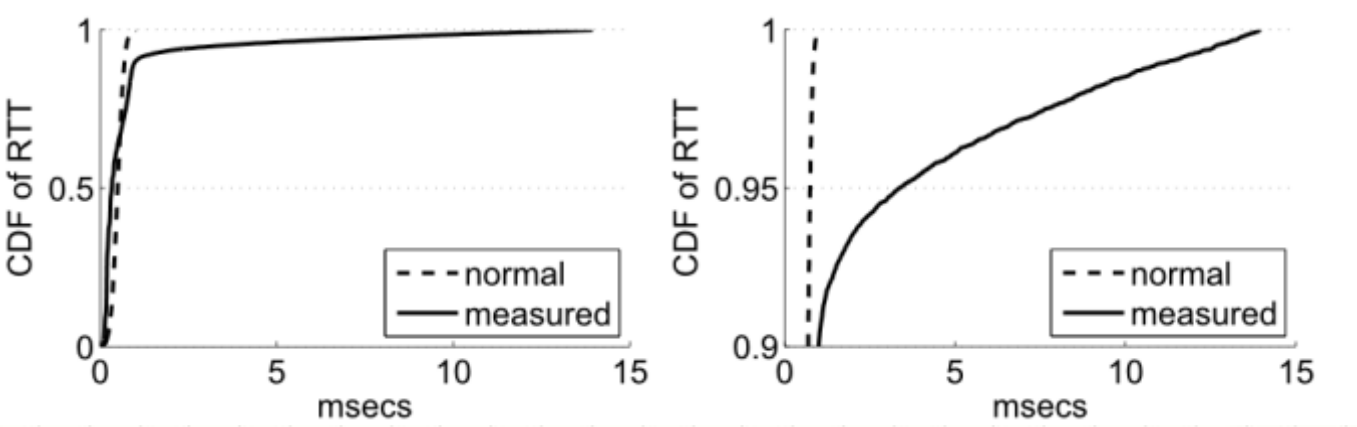
Contributions

1. Quantification of the impact of the long-tail flow completion times
2. Assessment of the causes of the long-tailed flow completion times
3. A cross-layer network stack that addresses them
4. Implementation-validated simulations demonstrating DeTail's significant improvement

Impact of the Long Tail

Traffic Measurements

- Intra-rack RTTs are typically low but congestion can cause them to vary by two orders of magnitude



Complete Distribution

90th – 100th Percentile

- The variation in RTTs is caused primarily by congestion

Impact on Workflows

Partition-Aggregate

- At the 99.9th percentile, a 40-worker flow has 4 workers (10%) miss their 10*ms* deadlines while a 400-worker flow has 14 (3.5%) miss theirs

Sequential

- At the 99.9th percentile, web sites must have less than 150 sequential data retrievals per page to meet 200*ms* page creation deadlines

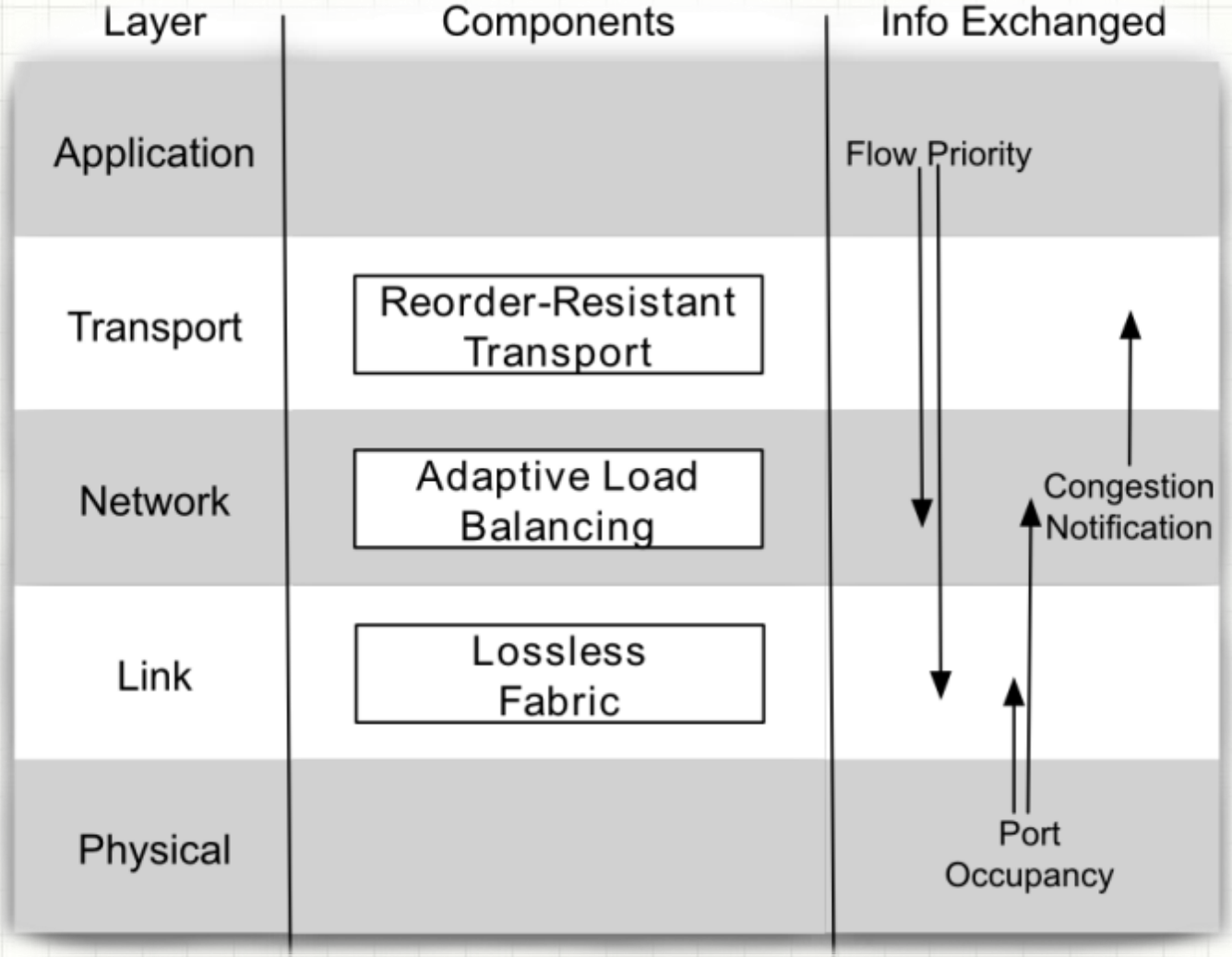
Based on published datacenter traffic measurements for production networks

While events at the long tail occur rarely, workflows use so many flows that several will experience delays for every page creation

A network that reduces the tail allows applications to render more complete pages without increasing server load

DeTail

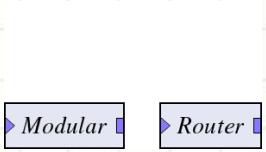
Cross-layer Network-based Approach



Simulation, Implementation and Experimental Results

Simulation and Implementation

- Simulation using CIOQ switch architecture in NS-3 Network Simulator
- NS-3 extended to include real-world processing delays
- NS-3 does not support ECN, but simulations still demonstrate impressive results



- Functional implementation using Click Modular Router
- Click software modified to have both ingress and egress queues
- Rate limiters added to prevent packet buildup in driver and hardware buffers

Experimental Results

To evaluate DeTail's ability to reduce the flow completion time tail, the following approaches are compared:

Flow Hashing (FH)

- Switches employ flow-level hashing
- Status quo and baseline

Lossless Packet Scatter (LPS)

- Switches employ packet scatter with PFC
- Not standard but can be deployed in current datacenters

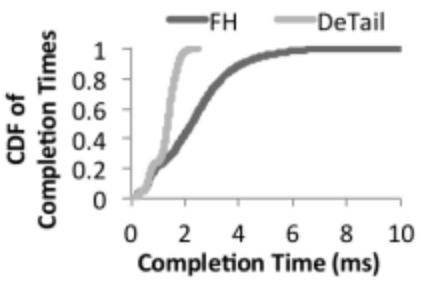
DeTail

- Switches employ PFC and Adaptive Load Balancing (ALB)
- New and exciting!

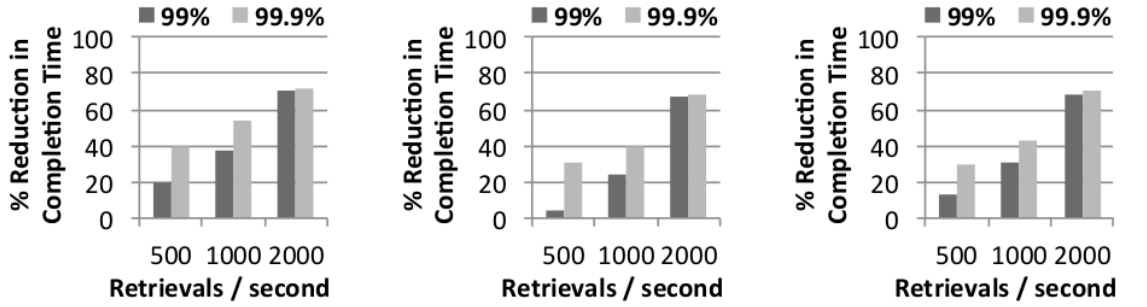
Simulator predictions are closely matched by implementation measurements!
The simulator is used to evaluate larger topologies and wider range of workflows

Microbenchmarks: All-to-All Workload

- FatTree topology with 128 servers in 4 pods with 4 ToR and 4 aggregate switches each
- Each server randomly retrieves data from another
- Servers also engaged in low-priority background flows



CDF of completion times of 8*KB* data retrievals at 2000 retrievals/second

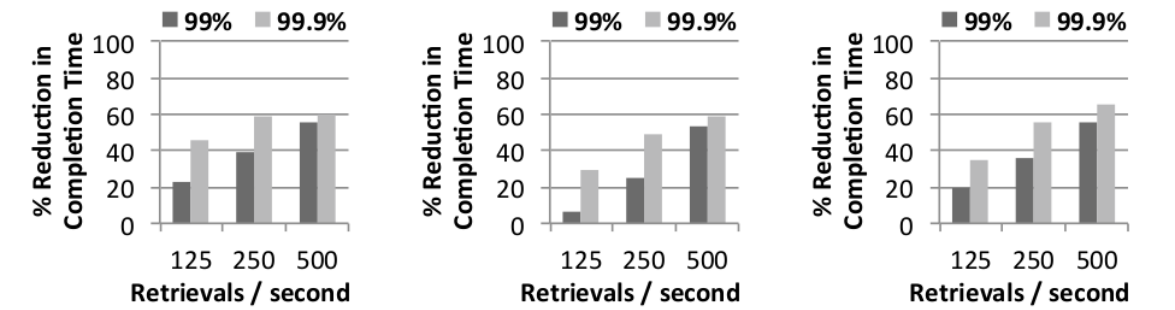


Reduction by DeTail over FH in 99th and 99.9th percentile completion times of 2*KB*, 8*KB* and 32*KB* retrievals.

DeTail provides up to 70% reduction at the 99th percentile.

Microbenchmarks: Front-end/Back-end Workload

- Same FatTree topology as before
- Servers in first three pods retrieve data from randomly chosen servers in fourth pod
- Servers also engaged in low-priority background flows

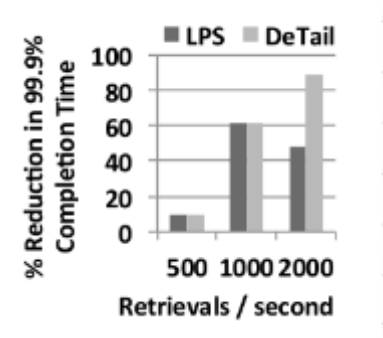


Reduction by DeTail over FH in 99th and 99.9th percentile completion times of 2*KB*, 8*KB* and 32*KB* retrievals. DeTail achieves 30% - 65% reduction in completion times at the 99.9th percentile.

Topological Asymmetries

Disconnected Link

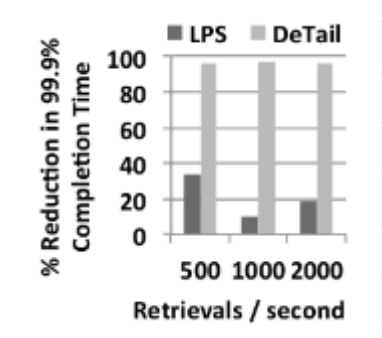
- Same as all-to-all workload but with one disconnected aggregate to core link



DeTail provides 10% - 89% reduction—almost an order of magnitude improvement—compared to FH for 8*KB* retrievals

Degraded Link

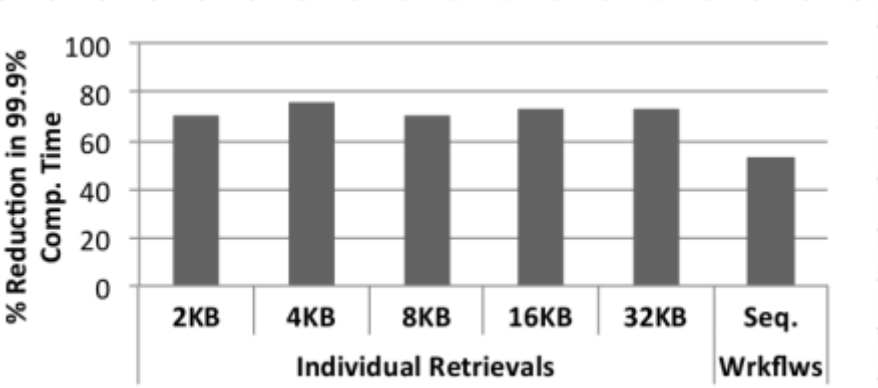
- Same as all-to-all workload but with one 1*Gpbs* downgraded to 100*Mbps*



DeTail provides 91% reduction compared to FH for 8*KB* retrievals

Web Workloads: Sequential

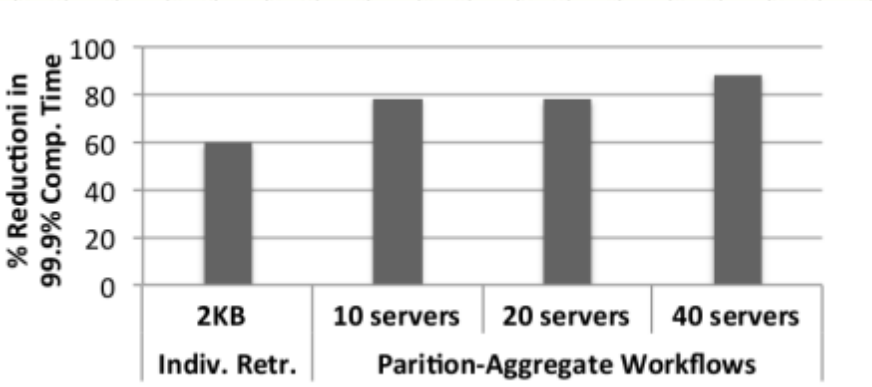
- Servers randomly assigned to be frond-end or back-end
- Front-end servers retrieve data from randomly chosen back-end servers
- Each sequential workflow consists of 10 sequential data retrievals of 2*KB*, 4*KB*, 8*KB*, 16*KB* or 32*KB*



DeTail provides 71% - 76% reduction in 99.9th percentile completion times of individual data retrievals and 54% reduction overall

Web Workloads: Partition-Aggregate

- Servers randomly assigned to be frond-end or back-end
- Front-end servers retrieve data in parallel from randomly chosen back-end servers
- Each partition-aggregate workflow consists of 10, 20, or 40 data retrievals 2*KB* in size



DeTail provides 78 - 88% reduction in 99.9th percentile completion times

Related Work and Summary

Related Work

Internet Protocols

- TCP Modifications: NewReno, Vegas, SACK
- Buffer Management: RED and Fair Queuing
- Operate at coarse-grained timescales inappropriate for datacenter workloads

Datacenter Networks

- Topologies: FatTrees, VL2, BCube, DCell
- Traffic Management: DCTCP, Hull, D^3, Datacenter Bridging
- Bound by performance of flow hashing

HPC Interconnects

- Credit-based flow control
- Adaptive Load Balancing: UGAL, PAR
- These mechanisms have not been evaluated for web-facing datacenter networks

Summary

DeTail is an approach for *reducing the tail completion times* of short, latency sensitive flows critical for page creation

DeTail employs cross-layer, in-network mechanisms to reduce packet losses, prioritize flows, and balance traffic

By making its flow completion statistics robust to congestion, DeTail can reduce 99.9th percentile flow completion times by over 50% for many workloads

QUESTIONS?

APPENDIX

Photo Credits

- Railroad crossing: Toledo Blade
  - http://www.toledoblade.com/frontpage/2008/03/04/Railroad-crossing-barriers-tested-in-Michigan.html
- Pin-the-Tail-on-the-Donkey: The City Patch
  - http://thecitypatch.com/2012/04/02/pin-the-tail-on-the-donkey-will-never-quite-be-the-same/
- Clip-art from Office.com