Measuring and Optimizing Tail Latency

Kathryn S McKinley, Google

CRA-W Undergraduate Town Hall
April 5th, 2018
**Speaker & Moderator**

**Kathryn S McKinley**

Dr. Kathryn S. McKinley is a Senior Research Scientist at Google and previously was a Researcher at Microsoft and an Endowed Professorship at The University of Texas at Austin. Her research spans programming languages, compilers, runtime systems, architecture, performance, and energy. She and her collaborators have produced several widely used tools: the DaCapo Java Benchmarks (30,000+ downloads), the TRIPS Compiler, Hoard memory manager, MMTk memory management toolkit, and the Immix garbage collector.

She served as program chair for ASPLOS, PACT, PLDI, ISMM, and CGO. She is currently a CRA and CRA-W Board member. Dr. McKinley was honored to testify to the House Science Committee (Feb. 14, 2013). She is an IEEE and ACM Fellow. She has graduated 22 PhD students.

**Lori Pollock**

Dr. Lori Pollock is a Professor in Computer and Information Sciences at University of Delaware. Her current research focuses on program analysis for building better software maintenance tools, software testing, energy-efficient software and computer science education. Dr. Pollock is an ACM Distinguished Scientist and was awarded the University of Delaware’s Excellence in Teaching Award and the E.A. Trabant Award for Women’s Equity.
Measuring and Optimizing Tail Latency

Kathryn S McKinley, Google

Xi Yang, Stephen M Blackburn, Md Haque, Sameh Elnikety, Yuxiong He, Ricardo Bianchini
Tail Latency Matters

400 millisecond delay decreased searches/user by 0.59%. [Jack Brutlag, Google]

Two second slowdown reduced revenue/user by 4.3%. [Eric Schurman, Bing]
## Datacenter economics quick facts*

| ~ $500,000  | Cost of small datacenter |
| ~ 3,000,000 | US datacenters in 2016   |
| ~ $1.5 trillion | US Capital investment to date |
| ~ $3,000,000,000 | KW dollars / year |
| ~ $30,000,000  | Savings from 1% less work |

Lots more by not building a datacenter

*Shehabi et al., United States Data Center Energy Usage Report, Lawrence Berkeley, 2016.*
Tail Latency

TOP PRIORITY

Efficiency

Google Cloud
Tail Latency

Efficiency

Both ?!
Server architecture

client

taggregator

workers
Characteristics of interactive services

Bursty, diurnal

CDF changes slowly

Slowest server dictates tail

Orders of magnitude diff

average & tail - 99th %tile
What is in the tail?
Cycle-level on-line profiling tool

[ISCA’15 (Top Picks HM), ATC’16]

Insight: Hardware & software generate signals without instrumentation

<table>
<thead>
<tr>
<th>counters</th>
<th>tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

- performance counters
- memory locations

HT1 IPC = Core IPC – HT2 SHIM IPC
What is in the tail?

![Graph showing latency distribution]

- Percentage of requests
- Latency (ms)
The Tail

Longest 200 requests

- Network & other
- Idle
- CPU work
- Queuing at worker

Network imperfections
OS imperfections
Long requests
Overload

noise
not noise
Optimizing the tail

Diagnosing the tail with continuous profiling

- **Noise** systems are not perfect
- **Queuing** too much load is bad, but so is over provisioning
- **Work** many requests are long

**Insights** Use the CDF off line

Long requests reveal themselves, treat them specially
Insight

Long requests reveal themselves

Regardless of the cause
Noise Replicate & reissue

The Tail at Scale, Dean & Barroso, CACM’13

![Graph showing latency and percentage of requests with noise, 5% and 10% reissued, CFD for cost & potential, and fixed issue time.]
Probabilistic Reissue

Adding randomness to reissue makes one earlier reissue time \( d \) (vs \( n \)) optimal.

Probability is proportional to reissue budget & noise in tail.

---

**Optimal Reissue Policies for Reducing Tail Latencies**, Kaler, He, & Elnickety, SPAA’17

---

[Diagram showing latency distribution and reissue probabilities]
Single R  Probabilistic reissue
Optimal Reissue Policies for Reducing Tail Latencies, Kaler, He, & Elnickety, SPAA’17
Work Speed up the tail efficiently

Judicious parallelism
[ASPLOS’15]

DVFS faster on the tail
[DISC’14, MICRO’17]

Asymmetric multicore
[DISC’14, MICRO’17]
Work Parallelism

Parallelism historically for **throughput**

**Idea** Parallelism for **tail latency**
Queuing theory

Optimizing average latency maximizes throughput

But not the tail!

Shortening the tail reduces queuing latency
Parallelism

Parallelism historically for **throughput**

**Idea** Parallelism for **tail latency**

**Insight** Long requests reveal themselves

**Approach** Incrementally add parallelism to long requests – the tail – based on request progress & load
Few to Many

**Fixed:** add thread every d ms

**Dynamic:** use load

- **Fixed:** add thread every d ms
  - Short delay good at low load
  - Long delay good at high load

- **Dynamic:** use load
  - Short delay good at low load
  - Long delay good at high load

---

**Diagram:***

- **Sequential**
- **4 way**
- **Fixed interval 20 ms**
- **Fixed interval 100 ms**
- **Fixed interval 500 ms**

Best at all loads
Evaluation 2x8 64 bit 2.3 GHz Xeon, 64 GB

Dynamic parallelism
Few to Many
21% fewer servers
or reduce tail by 28%
Work speed up the tail efficiently

Judicious parallelism
[ASPLOS’15]
Efficiency at scale for interactive workloads

Diagnosing the tail with continuous profiling

- **Noise** replication, systems are not perfect
- **Queuing** replication + judicious choice
- **Work** judicious use of resources on long requests

Request latency CDF is a powerful tool
Tail efficiency ≠ average or throughput
Hardware heterogeneity

Questions?
Professional and Research Relationships
Your Academic Village

- Peer students
- Students senior & junior to you
- Teaching assistants
- PhD students
- Faculty
My Professional Village

- Researchers in all career stages
  - Undergrads, PhD students, post docs
  - Faculty, industrial researchers, staff, administrators
- Industrial village
  - Software engineers in all career stages
  - Managers, directors, admins,
  - in/out my management chain
Faculty Mentors

Don Johnson
My Professor

Ken Kennedy
PhD Advisor

Dave Stemple
Dept. Chair
Building a Village
Networking is....

Building and sustaining professional relationships

- Participating in an academic / research community
- Finding people you like and you learn from, and building a relationship
Networking is not:

• Using people
• A substitute for quality work
But I am Horrible at Small Talk

• You have CS in common
• Networking is not genetic
• It is a research skill
  – Practice
  – Meet people
  – Learn
  – Go places
  – Volunteer!
• Sustain your relationships
With whom do you network?

- People you like
- People senior to you, who can show you the way
- People at different career stages, so you can anticipate
- Your peers
Peer Mentors

Mary Hall

Doug Burger

Margaret Martonosi
Your Village Will

- Write letters for grad school, jobs, etc.
- Help you solve problems
- Point you in good directions
- Encourage you
- Choose you for important roles
- You will do the same or more for them
- Make your life and work more fun and meaningful