Cluster management at Google

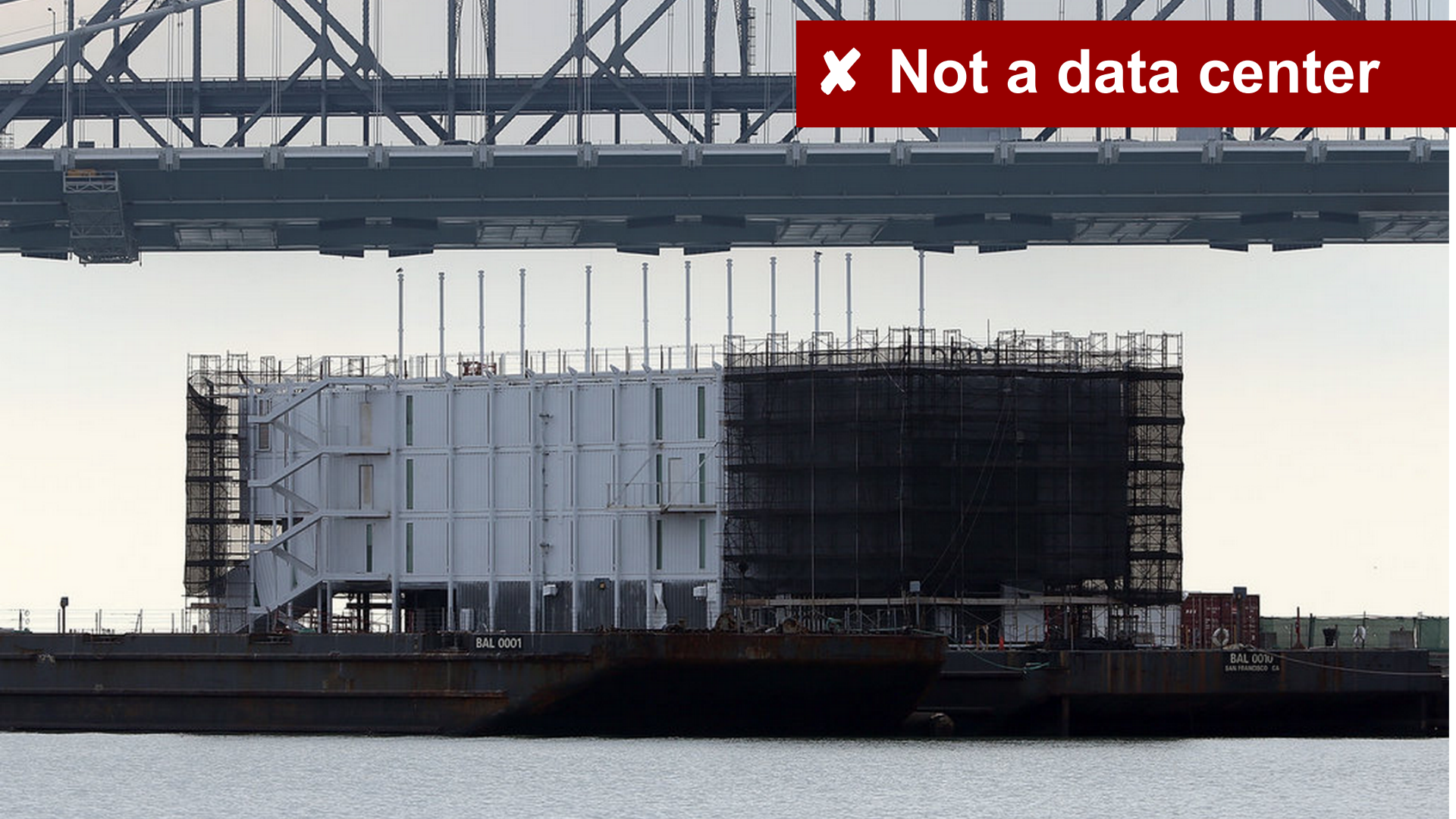
LISA 2013

john wilkes (johnwilkes@google.com)
Software Engineer, Google, Inc.

Google

We own and operate data centers around the world

http://www.google.com/about/datacenters/inside/locations/


✘ Not a data center
BAL 0001
BAL 0010
SAN FRANCISCO CA

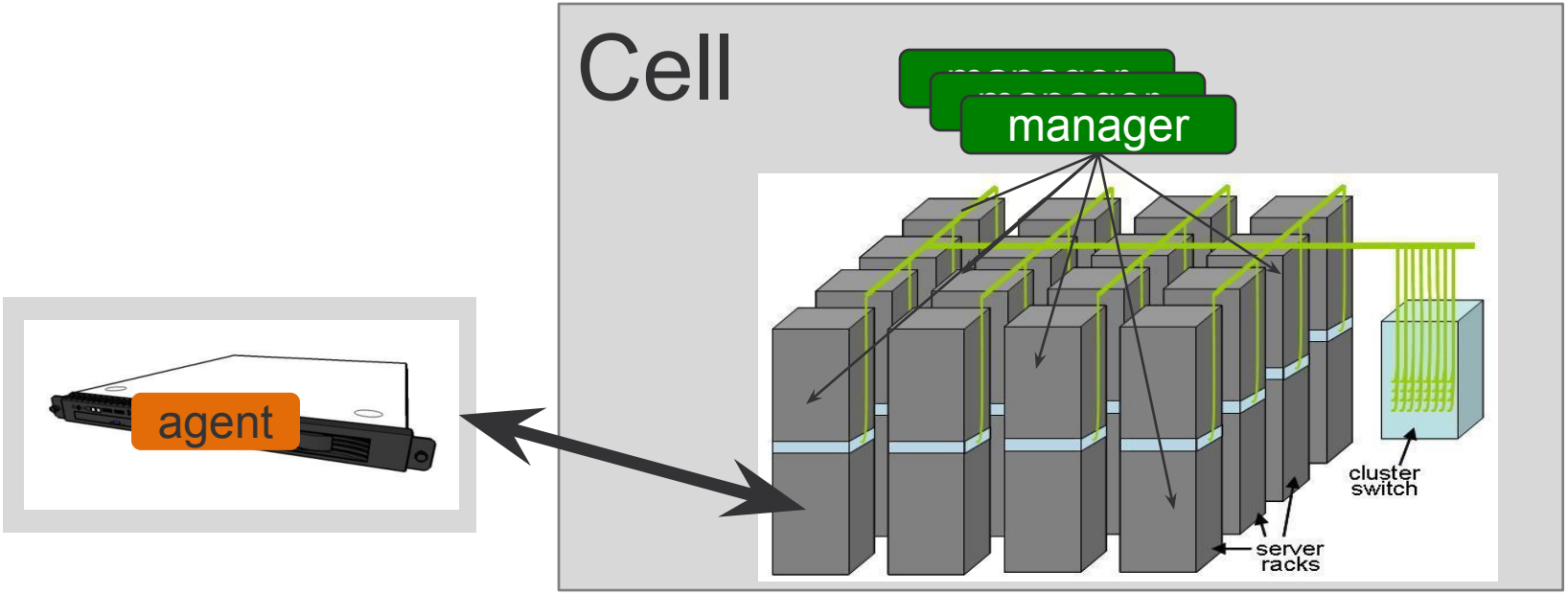

✔ Data center
Google | google.com/datacenters


Google | google.com/datacenters


Google | google.com/datacenters

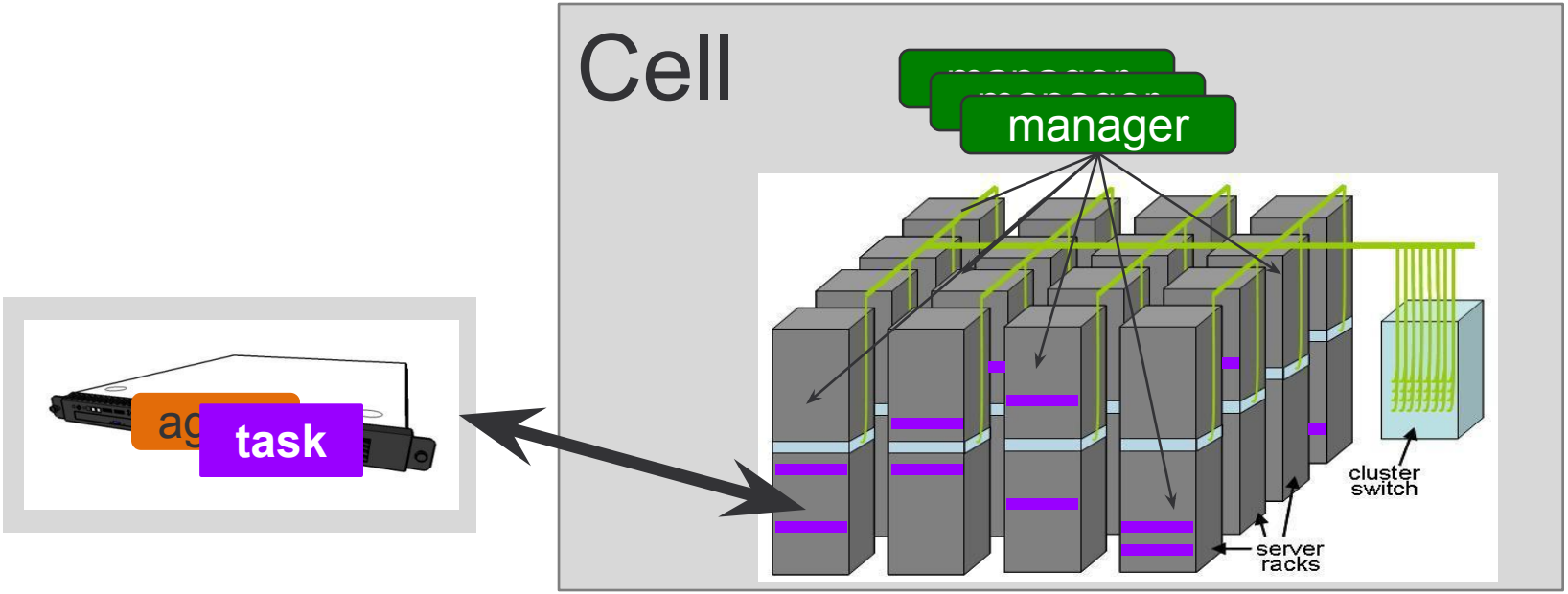
Cluster management: *what is it?*

A cluster is managed as 1 or more **cells**



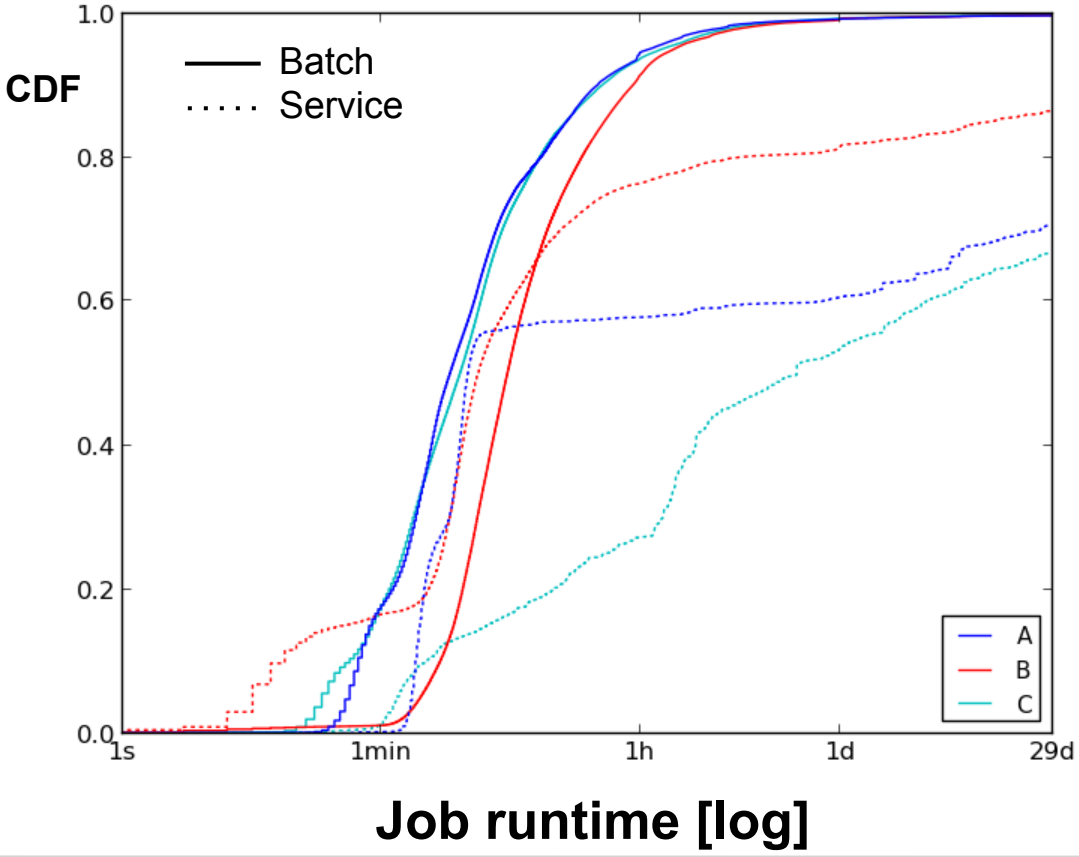
Cluster management: *what is it?*

A cell runs **jobs**, made up of (1 to thousands) of **tasks**, for internal users



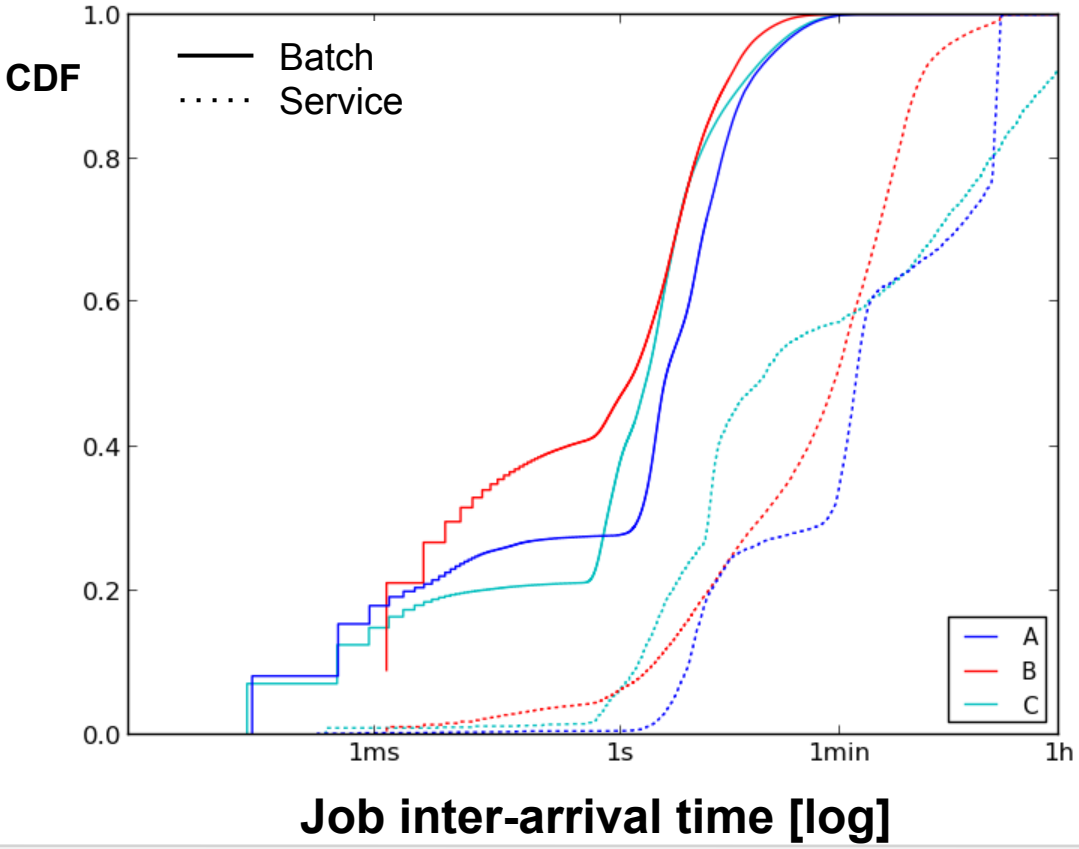
Google

workload: job runtimes



Google

workload: job inter-arrival times



workload: batch/service split

Cluster A
Medium size

Medium utilization

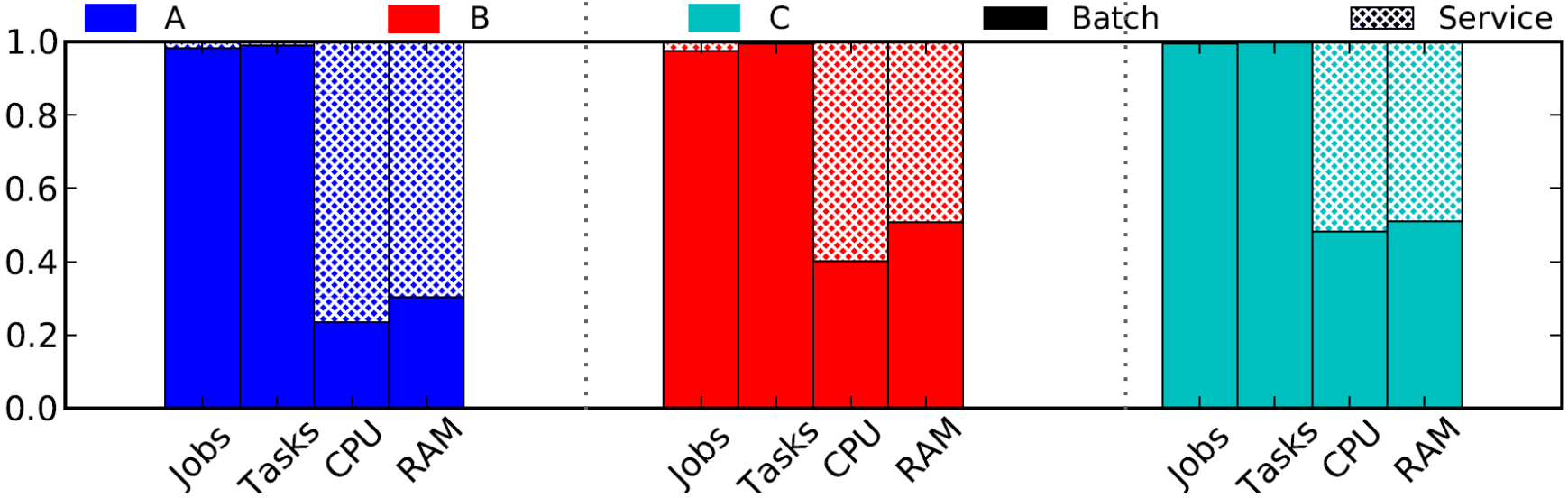
Cluster B
Large size

Medium utilization

Cluster C
Medium (12k mach.)

High utilization

Public trace

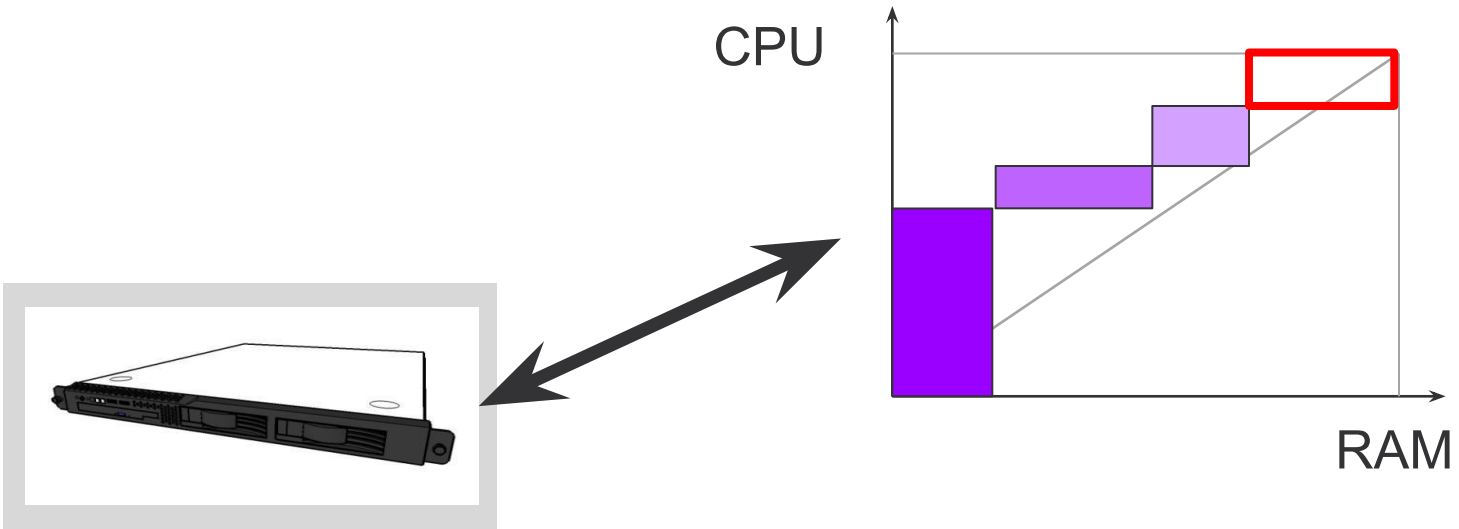


Jobs/tasks: counts

CPU/RAM: resource seconds [i.e. resource job runtime in sec.]

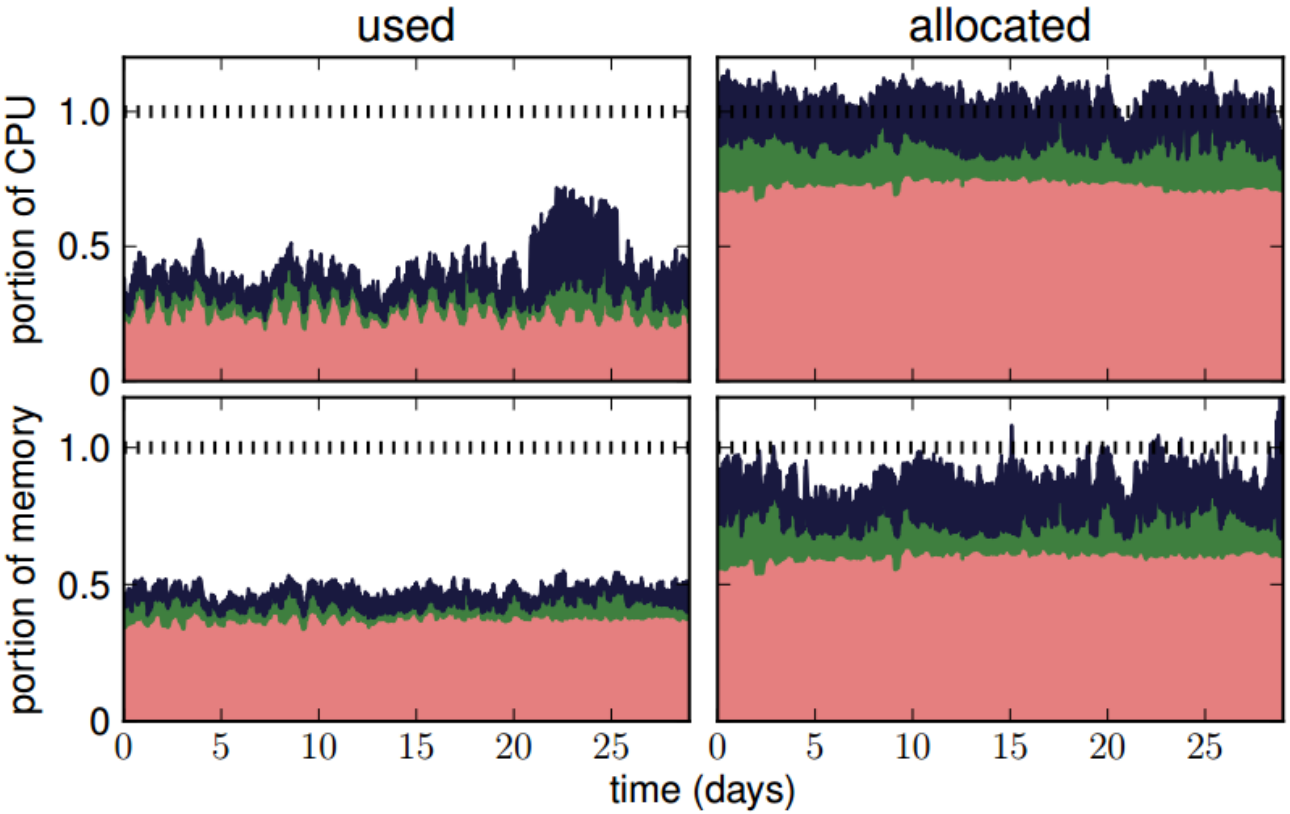
Cluster management: *what is it?*

Placing tasks onto machines is just a *knapsack problem*, with constraints



Google

Cluster management: *what is it?*



Heterogeneity and dynamicity of clouds at scale: Google trace analysis. SoCC'12

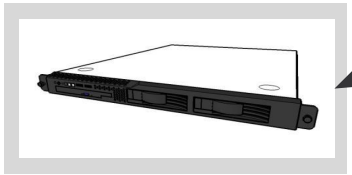
public cell workload trace

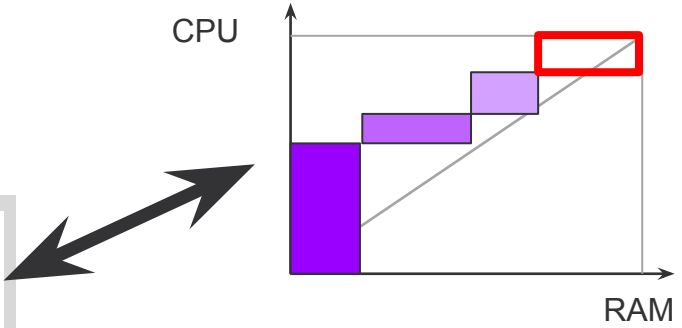
Want to try it yourself?

- search for [john wilkes google cluster data]

29 day cell workload trace from May 2011

- ~12.5k machines
- every job/task/machine event
- task usage every 5 mins
- (obfuscated data)






Ich bin die ewig

a few other moving parts

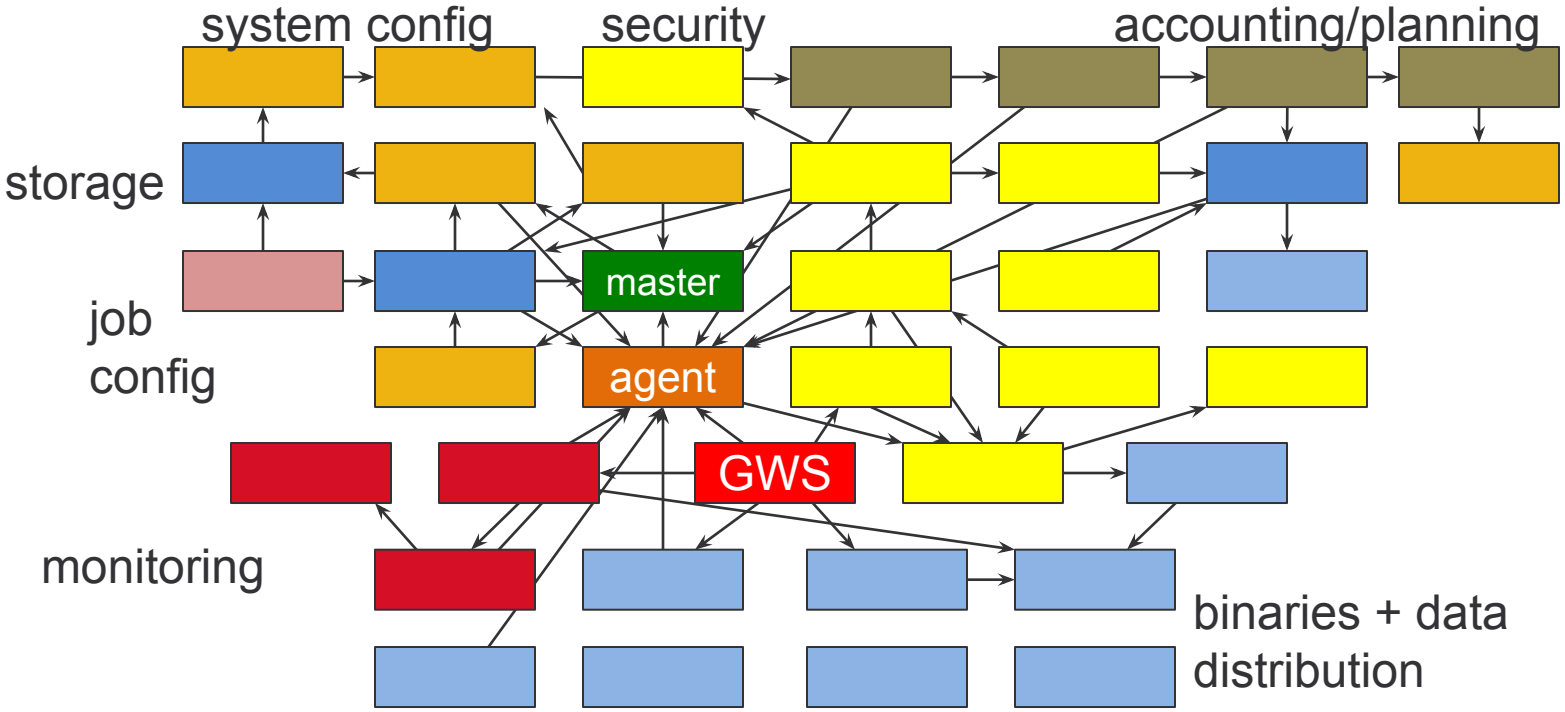


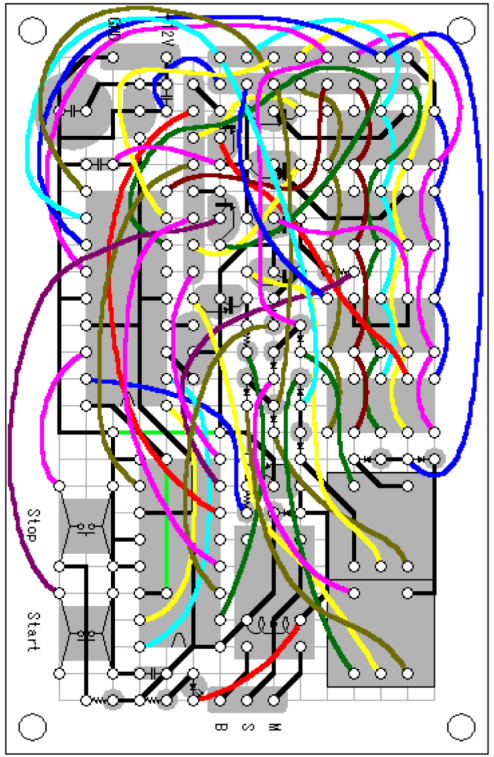
Diagram from an original by Cody Smith.

Make an app work right once: ***simple***

Google Docs uses ~50 systems and services

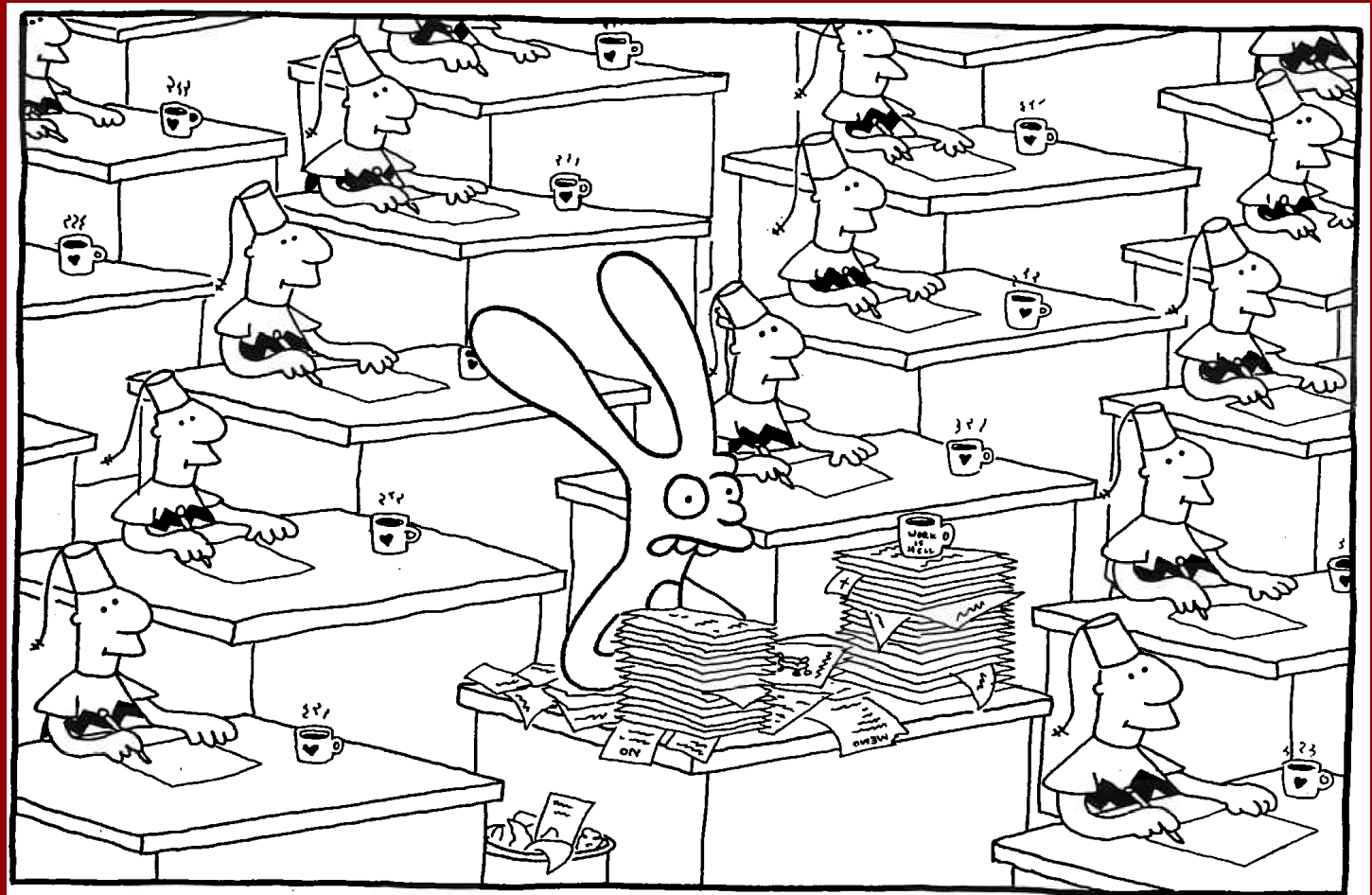
Make it run in production: ***priceless***

- run it in 6 cells
- fix it in an emergency
- move it to another cell



http://melinathinks.com

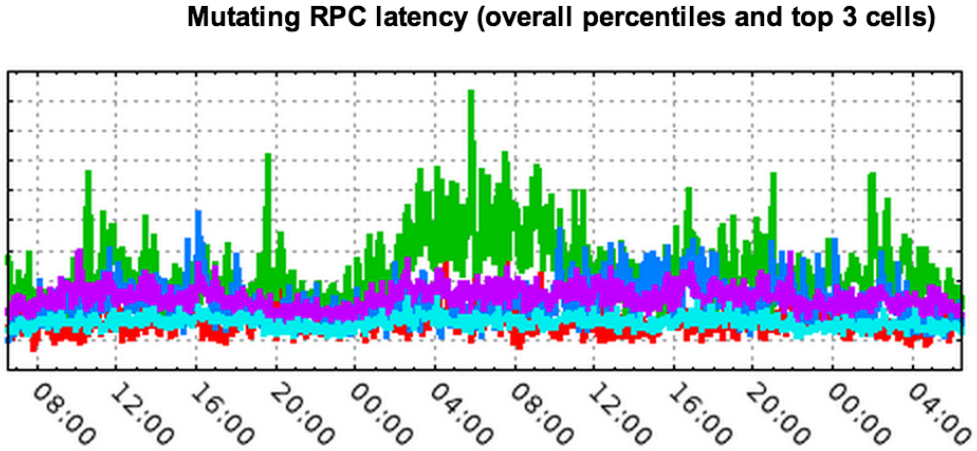
If you aren't measuring it, it is out of control ...



SLI/SLOs for cluster manager

1. Availability
   - % time master is up
   - max outage duration

2. Performance
   - RPC request latencies
   - Time to schedule+start job



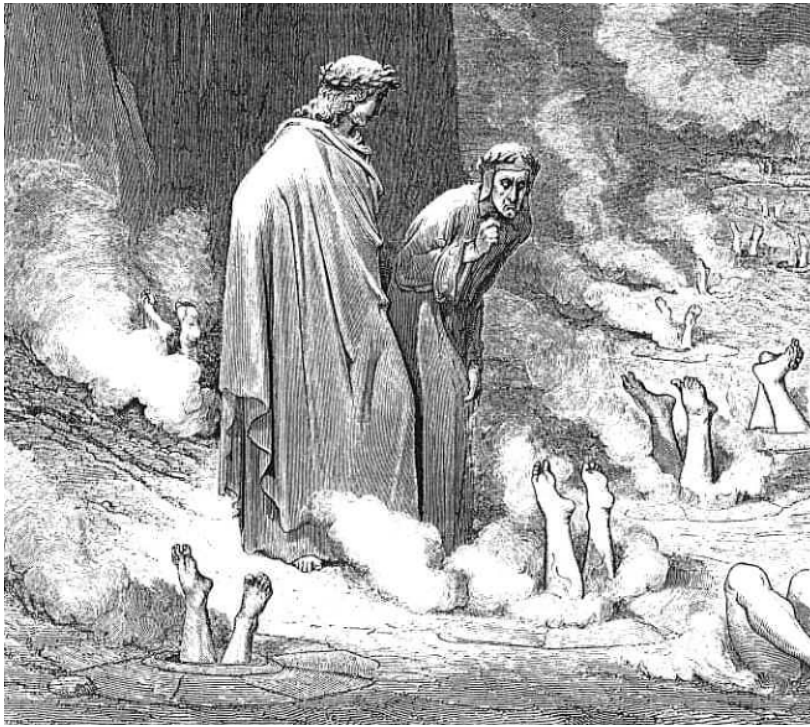
3. Evictions
   - % jobs with too high an eviction rate
   - % jobs with too many concurrent evictions
   - % tasks evicted without proper notice

Cluster management: *pre-Omega*

The current system was built 2003-4. It works pretty well ☺

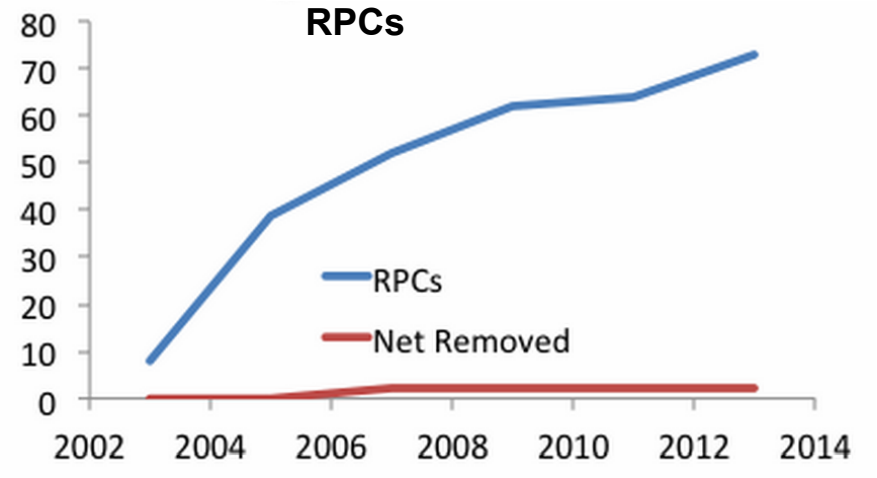
But ...

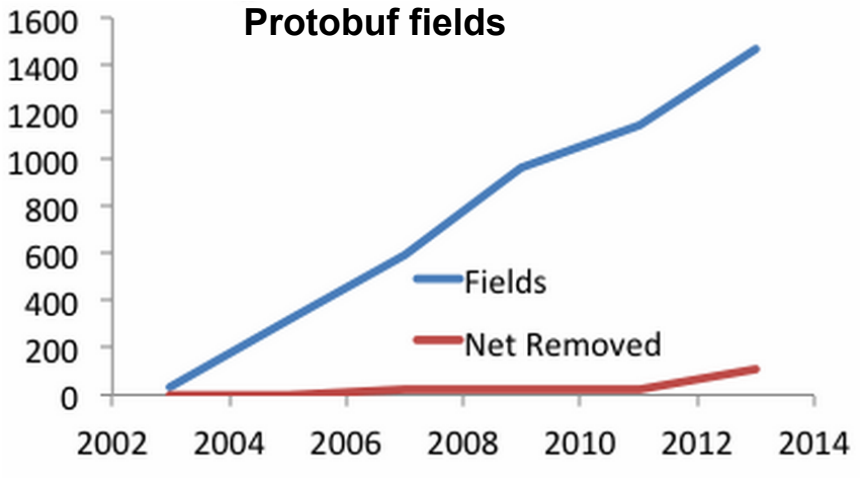
- growing scale (#cores)
- inflexibility (adding features)
- internal complexity (adding people)



Google

Cluster management: *pre-Omega*



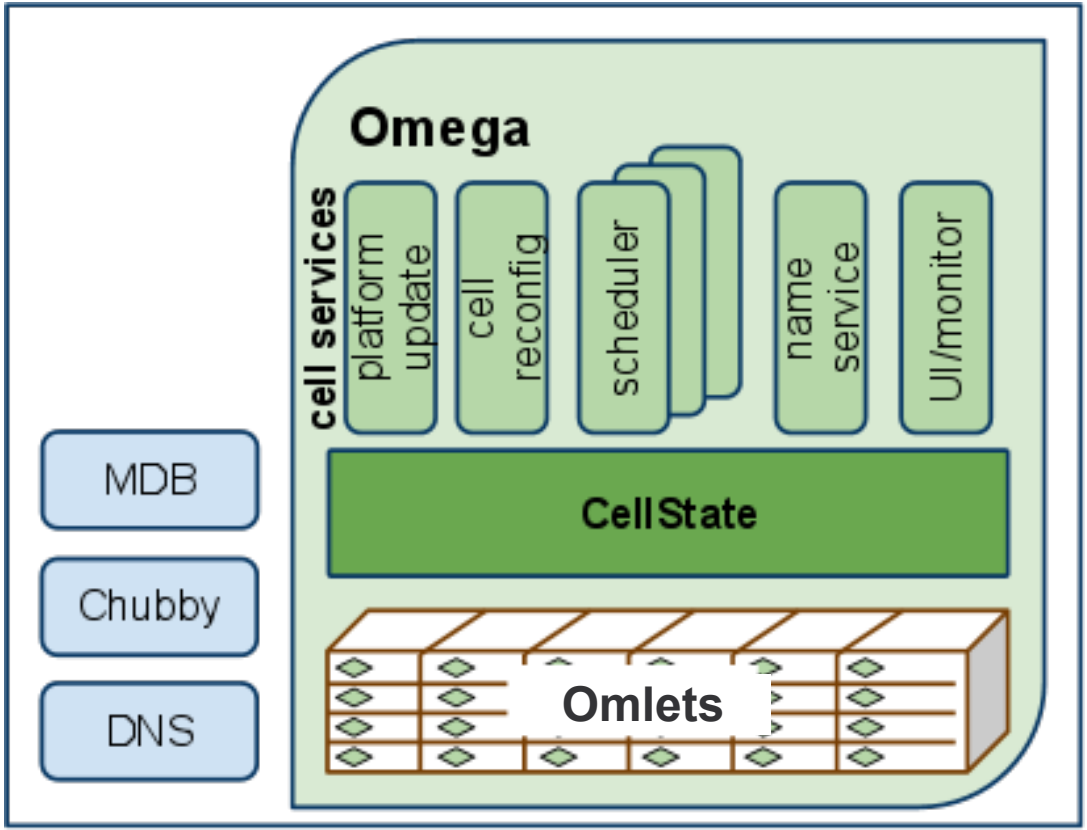


The “second system” …

- Main *user* goals: predictability & ease of use
- Main *team* goal: flexibility

Omega is currently being deployed + prototyped

overall architecture

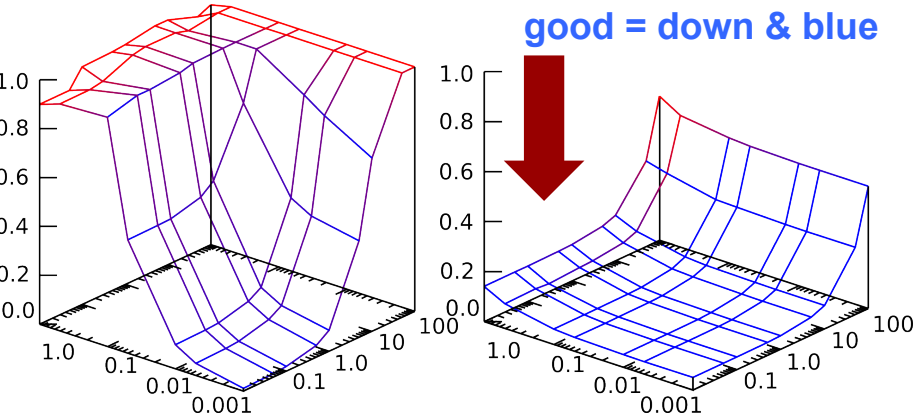


CellState:

- minimum necessary data
- no policies
- just enforces invariants

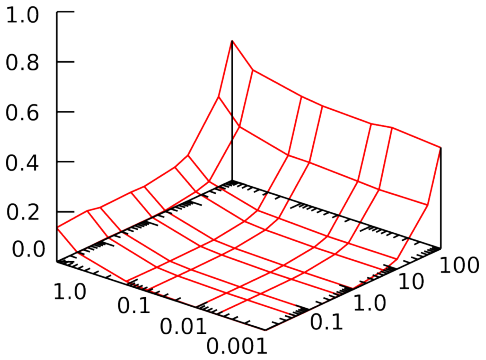
Calendar: <u>everything</u> is a scheduling event!

performance: simulation study

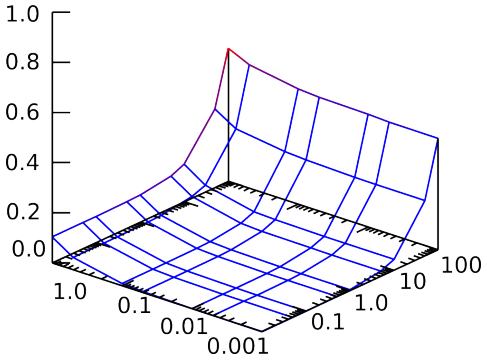


one path fast batch path

Monolithic scheduler



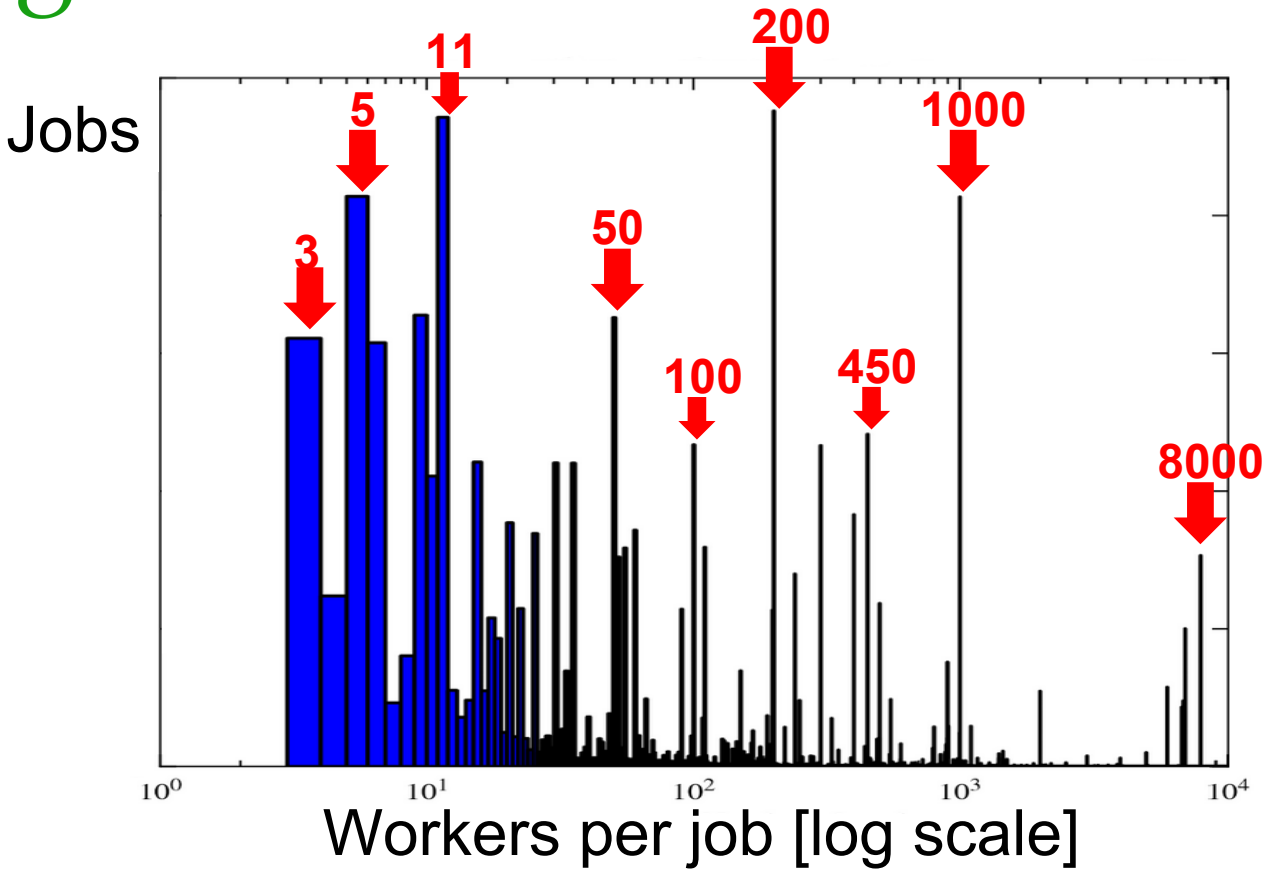
UCB Mesos



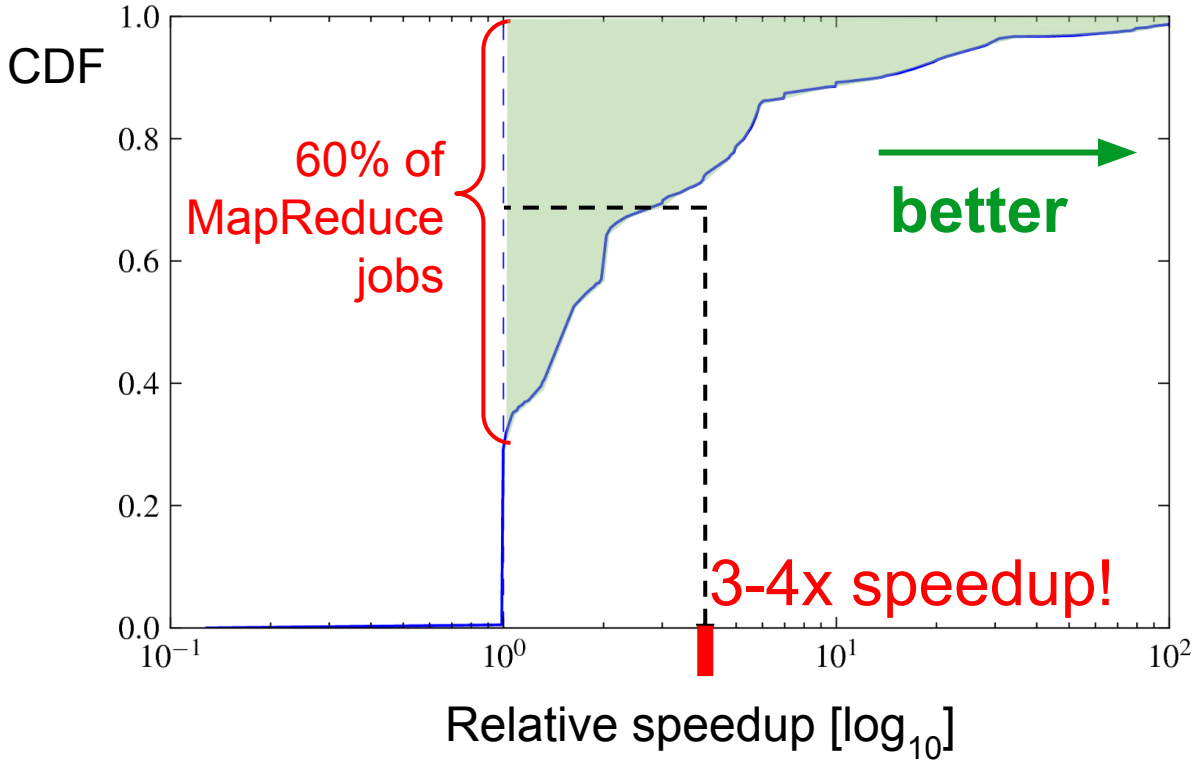
Omega

Ωmega

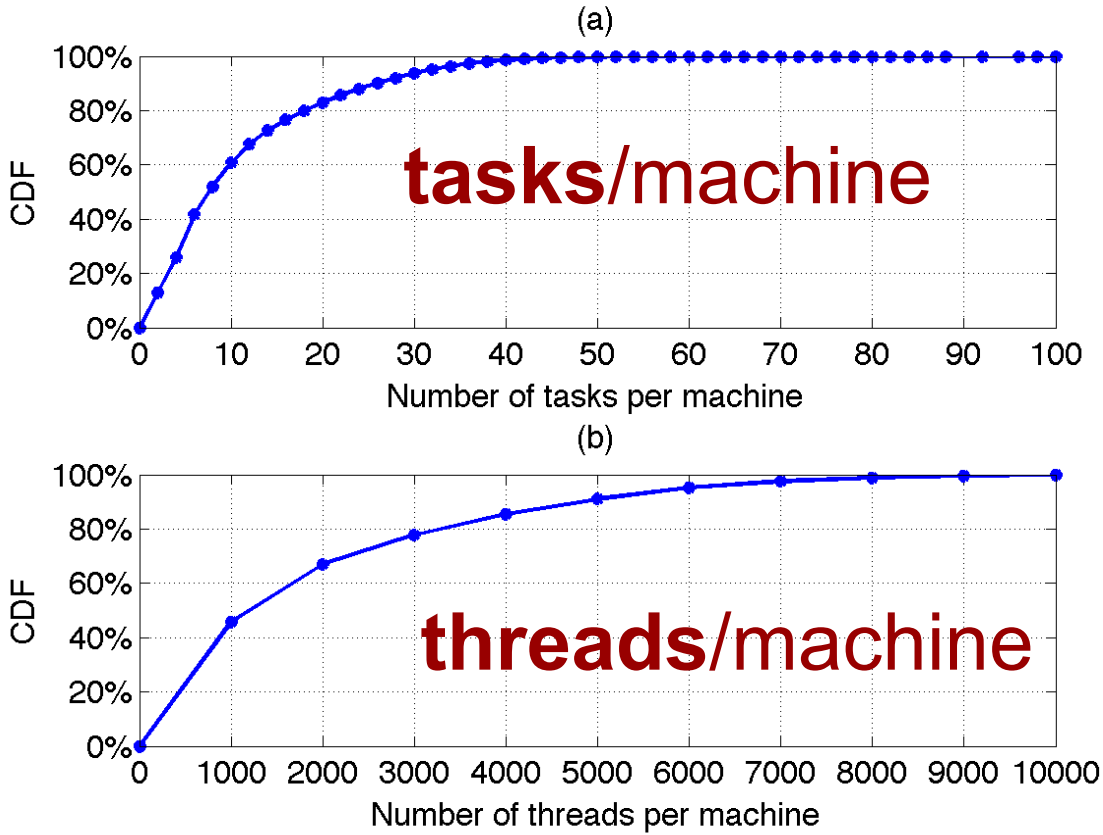
flexibility: MapReduce scheduler



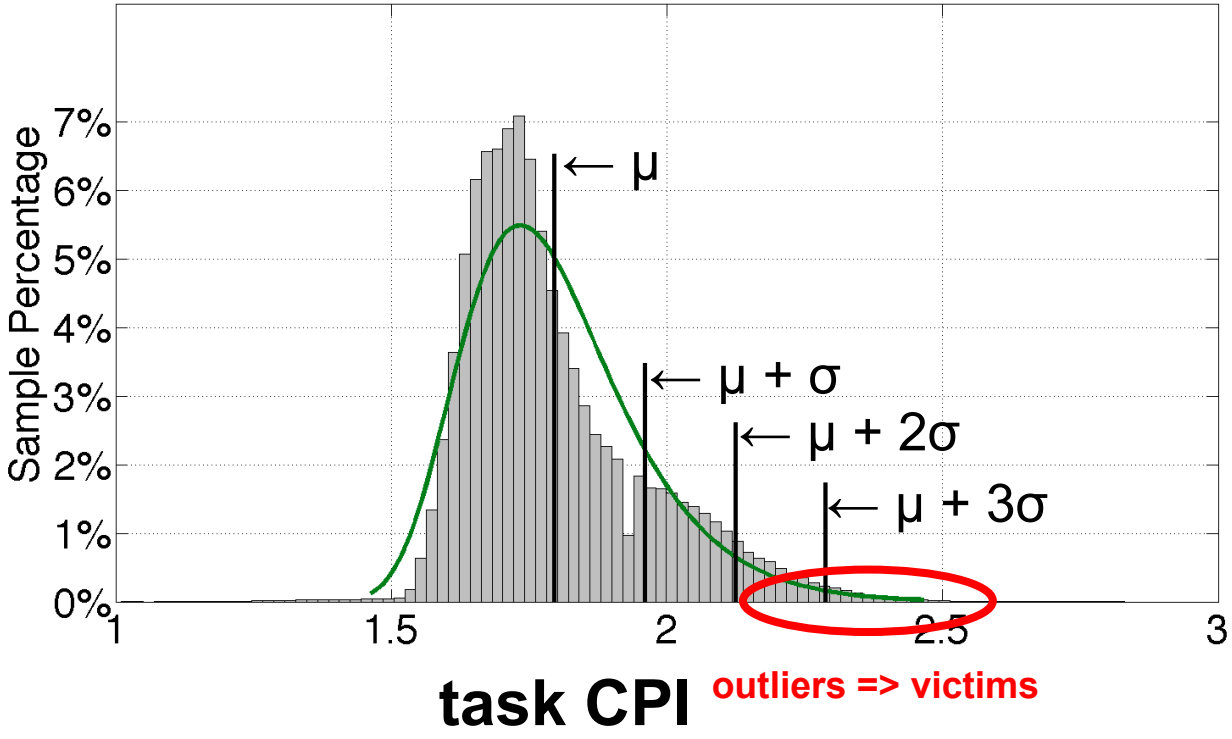
flexibility: MapReduce scheduler



multiple applications per machine

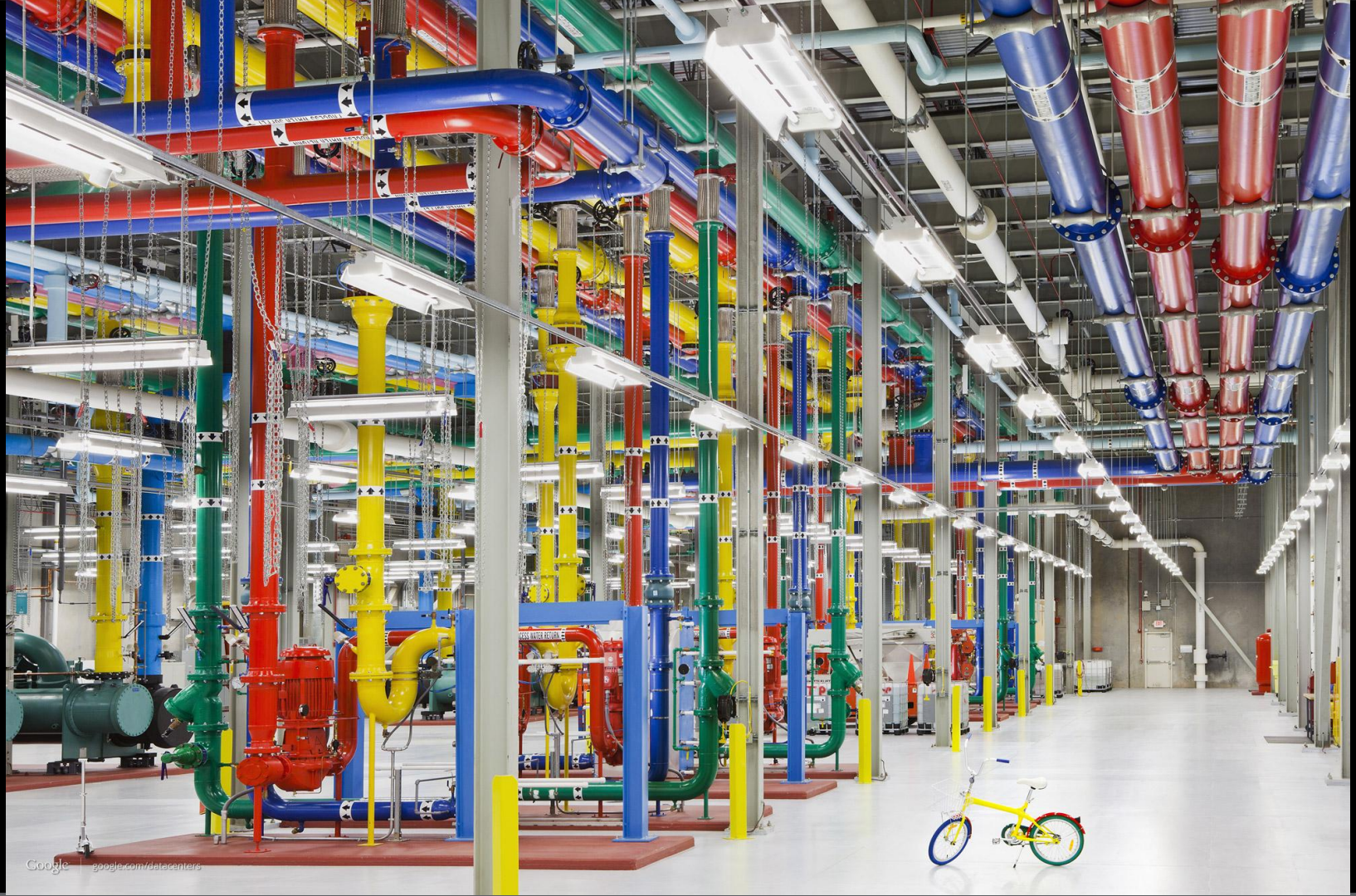


multiple applications per machine



CPI data for all the tasks in a job

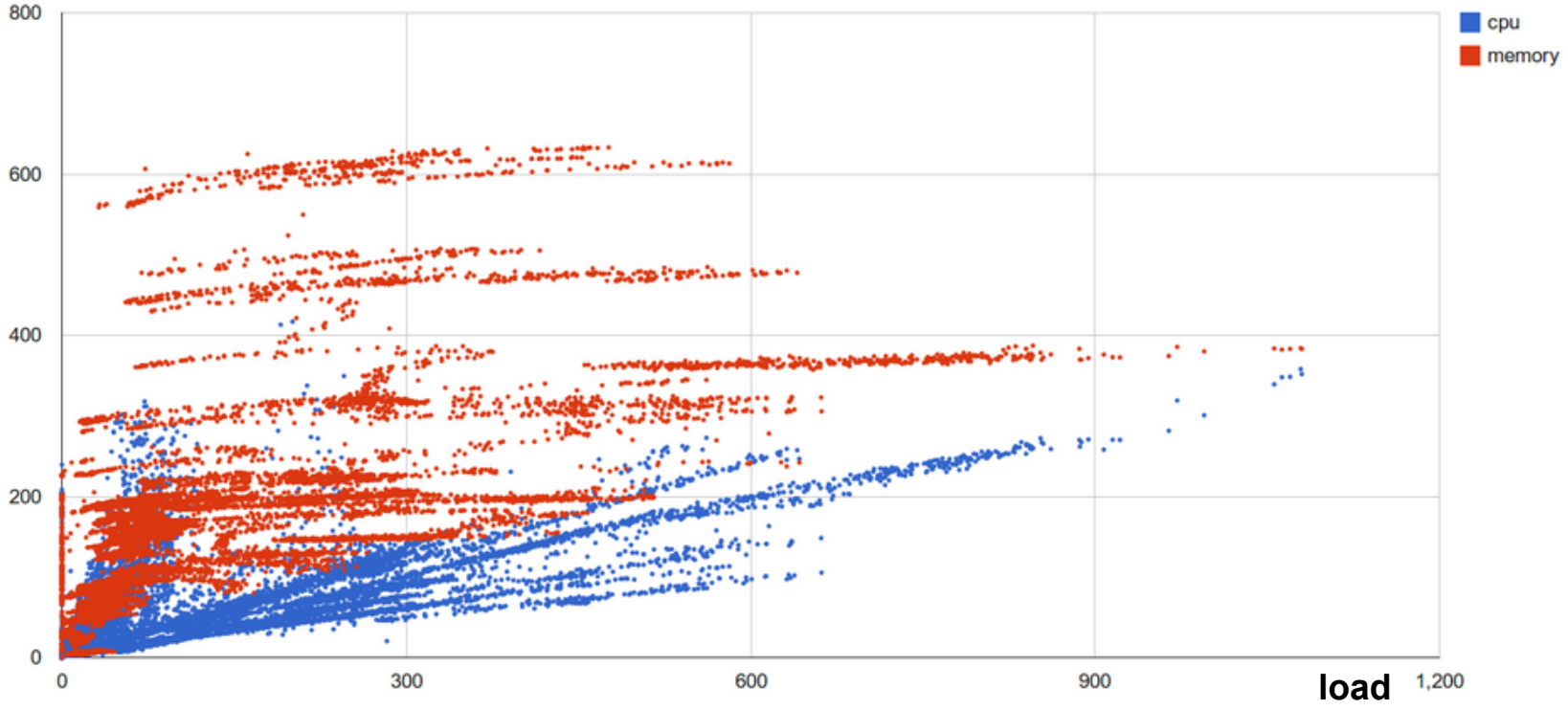
- per-cluster
- per-platform
- mean (μ)
- stddev (σ)


Google
google.com/datacenters

Omega goals: *ease of use*

Workload variation

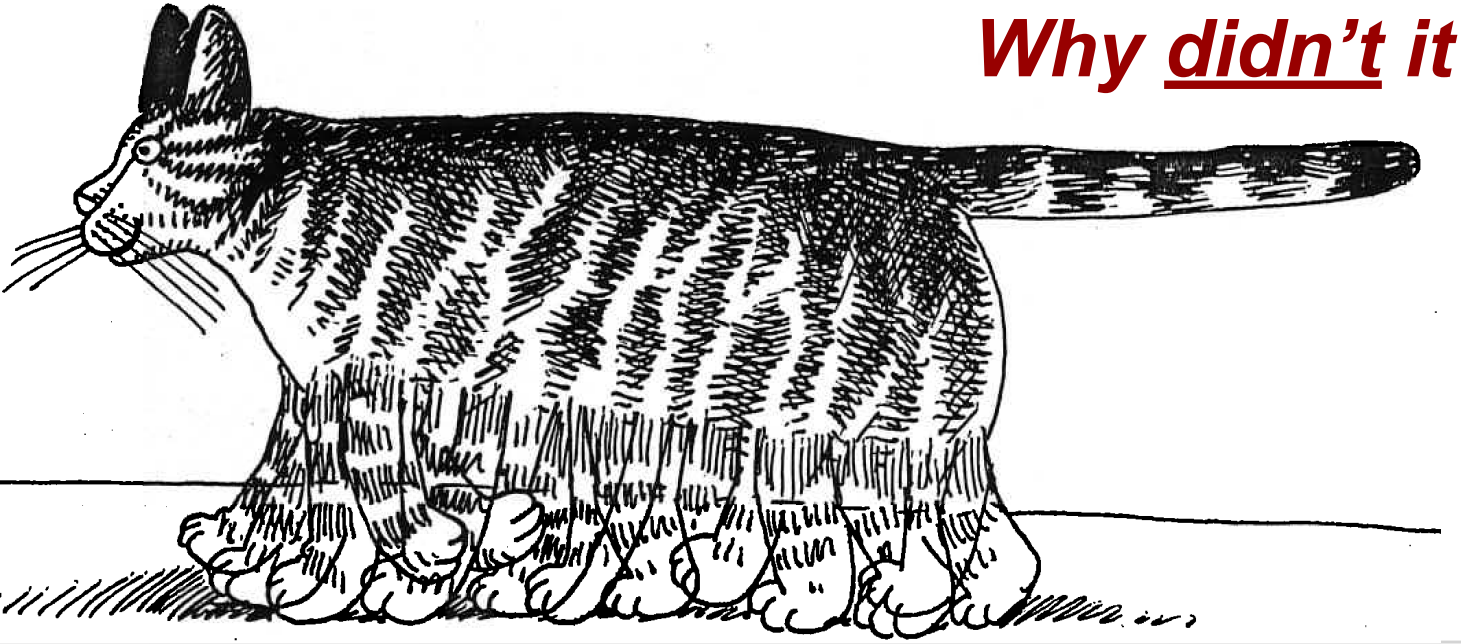
CPU and RAM usage as a function of offered load



Omega goals: *ease of use*

"Here's a binary … run it"

Why didn't it run?



Open issues: *smart explanations*

It's 3am and your pager goes off

- are we in trouble?
- are we about to get into trouble?

➔ what should you do about it?

Large-scale systems are exciting beasts

- scale → failures
- QoS → new designs
- configuration may be **the** next big challenge

We're hiring! www.google.com/jobs

johnwilkes@google.com