Pragmatic Performance

@giltene

Gil Tene, CTO & co-Founder, Azul Systems

AZUL SYSTEMS

About me: Gil Tene

- co-founder, CTO @Azul Systems
- Have been working on "think different" GC approaches since 2002
- A Long history building Virtual & Physical Machines, Operating Systems, Enterprise apps, etc...
- I have made a lot of mistakes
- I learned from some of those...

* working on real-world trash compaction issues, circa 2004

AZUL SYSTEMS

Pragmatic Performance

@giltene

Gil Tene, CTO & co-Founder, Azul Systems

AZUL SYSTEMS

prag·mat·ic

/pragˈmadik/

adjective

dealing with things sensibly and realistically in a way that is based on *practical* rather than *theoretical* considerations.

AZUL SYSTEMS

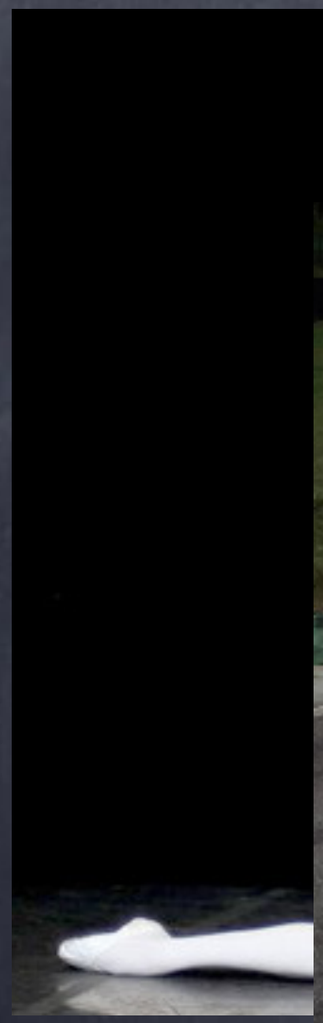
Performance

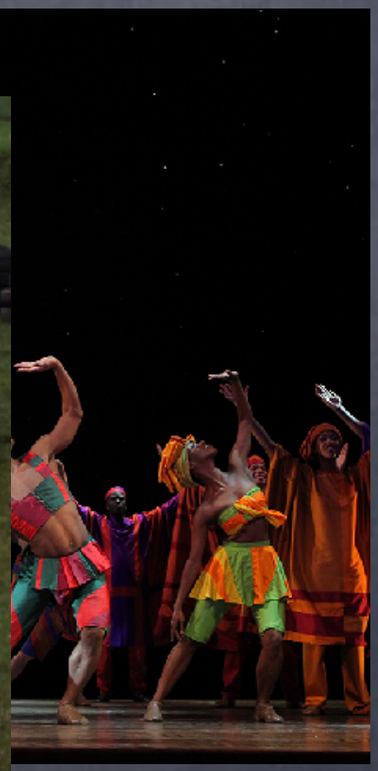
AZUL SYSTEMS®

Performance

AZUL SYSTEMS®

Performance





AZUL SYSTEMS

per·for·mance

/pərˈfôrməns/

noun

1. an act of staging or presenting a play, concert, or other form of entertainment

2. the action or process of carrying out or *accomplishing* an *action*, *task*, or *function*.

AZUL SYSTEMS

Performance

AZUL SYSTEMS®

Theoretical Performance

Question: How fast can this car go?

AZUL SYSTEMS®

Theoretical Performance

Question: How fast can this car go?

Theoretical Performance answer: 189mph

AZUL SYSTEMS

Theoretical Performance

Faster?

Slower?

AZUL SYSTEMS

Pragmatic Performance

Faster?

Slower?

How about now?

AZUL SYSTEMS

Pragmatic Performance

Pragmatic Question: How fast can this car go without crashing into things?

AZUL SYSTEMS®

Theoretical Performance

How many queries per second
can "cool tool X" answer?

AZUL SYSTEMS

Theoretical Performance

How many queries per second can "cool tool X" answer?

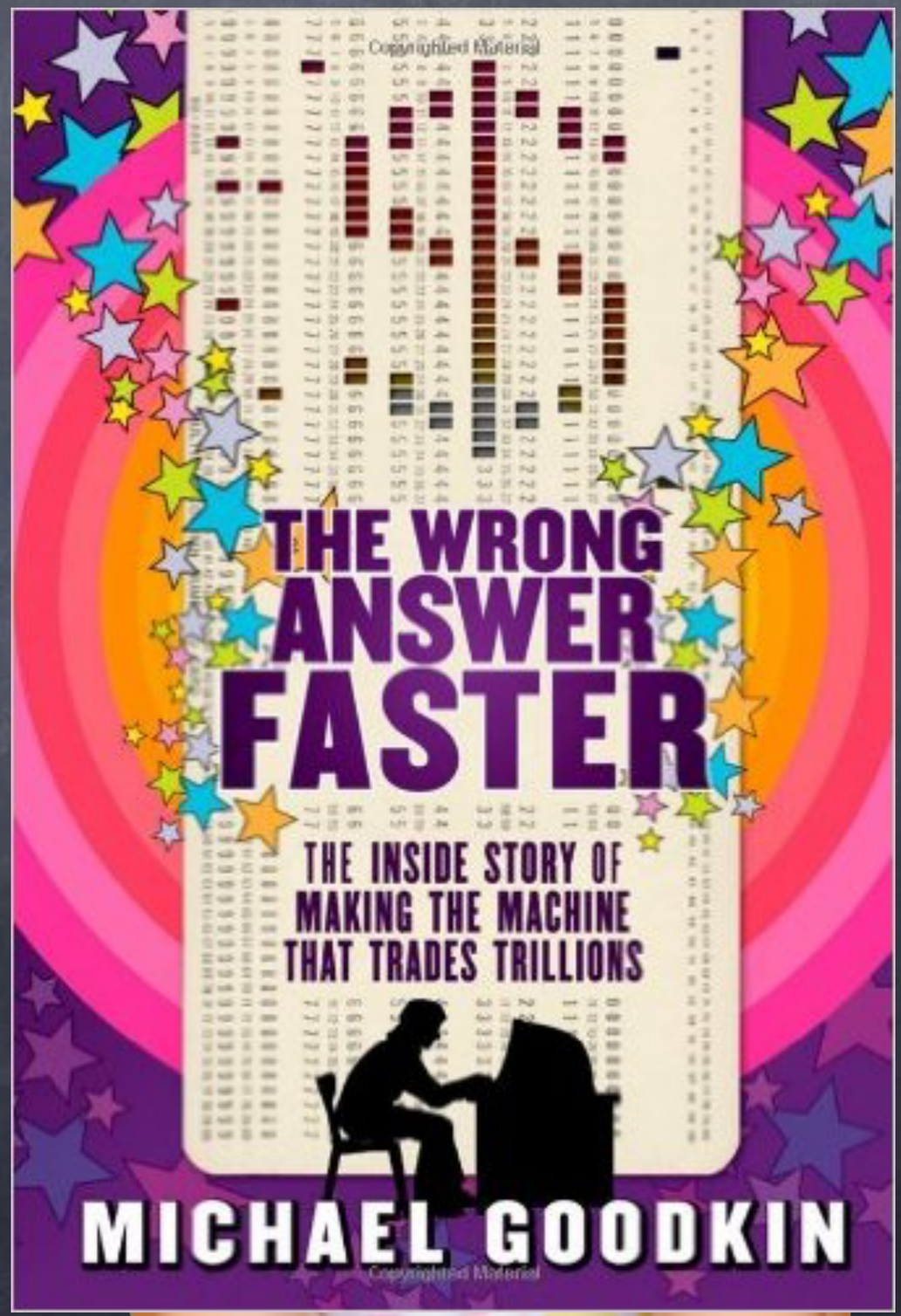
Examples of important questions to ask:

- Do all the answers have to be correct?
- Is it OK to only answer easy questions?
- Is it OK to take 1,000,000,000,000 questions now and answer them all next week?

AZUL SYSTEMS

Comparing Performance

- System A does X things per second. But fails some key requirements
- System B does 0.9x things per second, and meets all key requirements
- "System B is slower but more reliable" — WRONG
- How fast can system A go while meeting requirements?



AZUL SYSTEMS

Performance does not live in a vacuum

Performance metrics are (usually) meaningful only when practical considerations and constraints are met

AZUL SYSTEMS

performance metric examples

- Operations per second
- Latency or Response Time
- Failure rate
- Recovery time (e.g. to "normal") after disruption
- Each of these is best measured when all the others are held to required levels

AZUL SYSTEMS

performance metric examples

- Operations per second ("speed", "throughput")
- Latency or Response Time ("quickness")
- Failure rate ("reliability", "availability")
- Recovery time (e.g. to "normal") after disruption
- Each of these is best measured when all the others are held to required levels

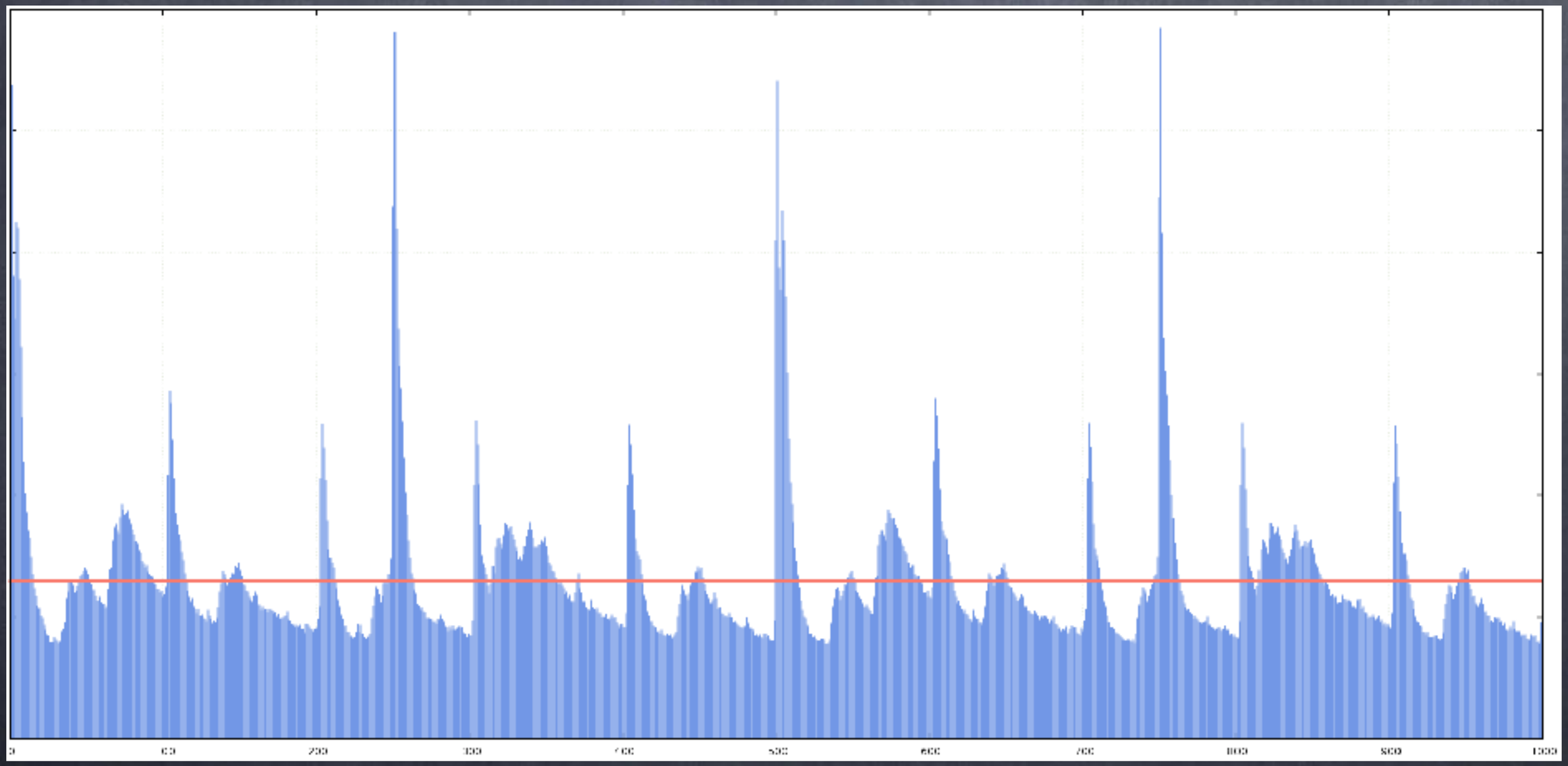
AZUL SYSTEMS

Capacity planning:
How much "capacity" do we need?

- A system takes 200 usec to respond to a request
- The system will receive 250 req/sec during peaks
- Requirement: 99.9% of operations must complete in 10 msec of less, even during busiest second
- Does the system have the capacity to handle the the load with the required behavior?
- Maybe…
- Not enough information…

AZUL SYSTEMS®

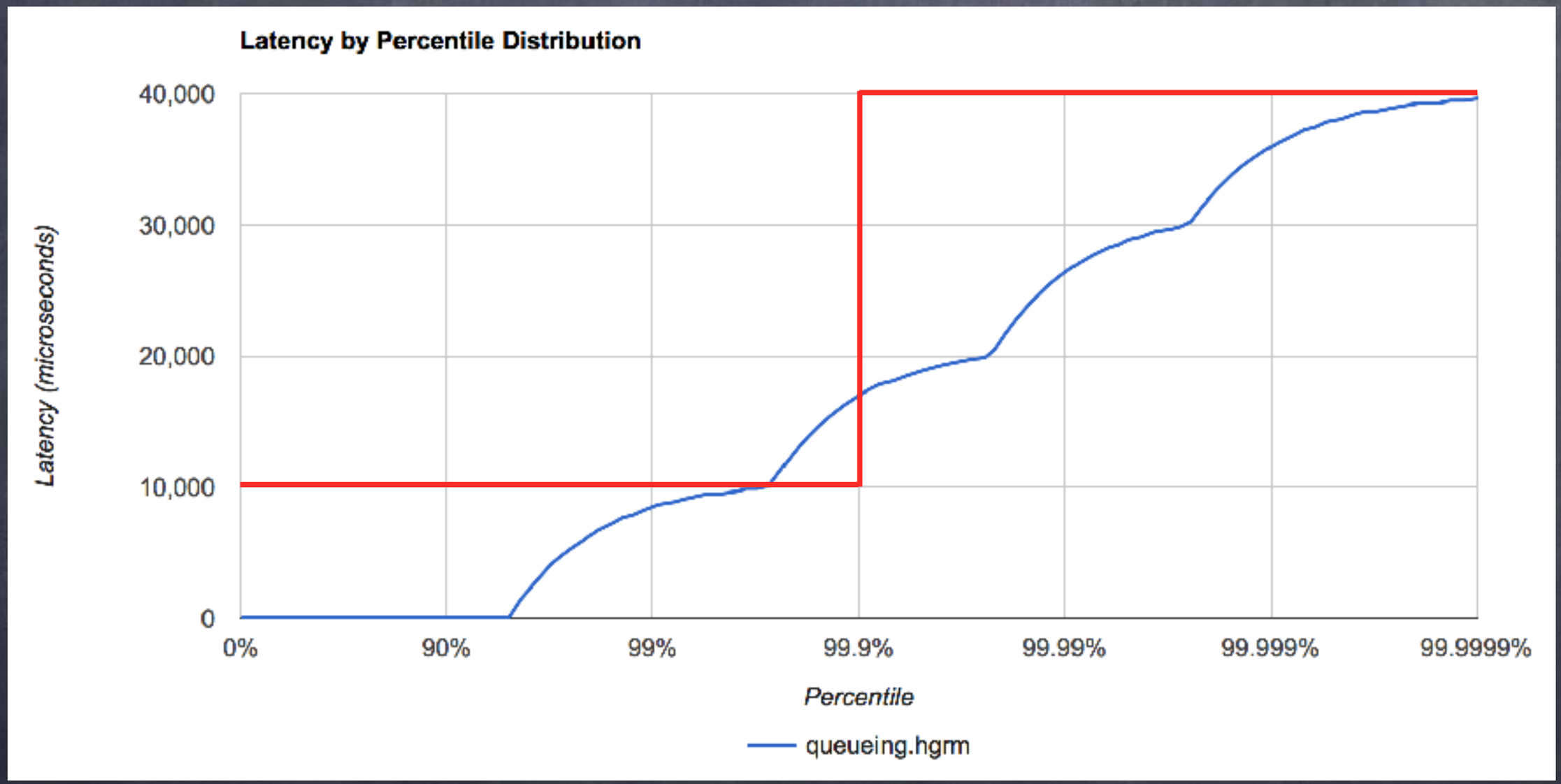
Arrival times



Example: arrival rate within a second
(averaged over entire day)

AZUL SYSTEMS

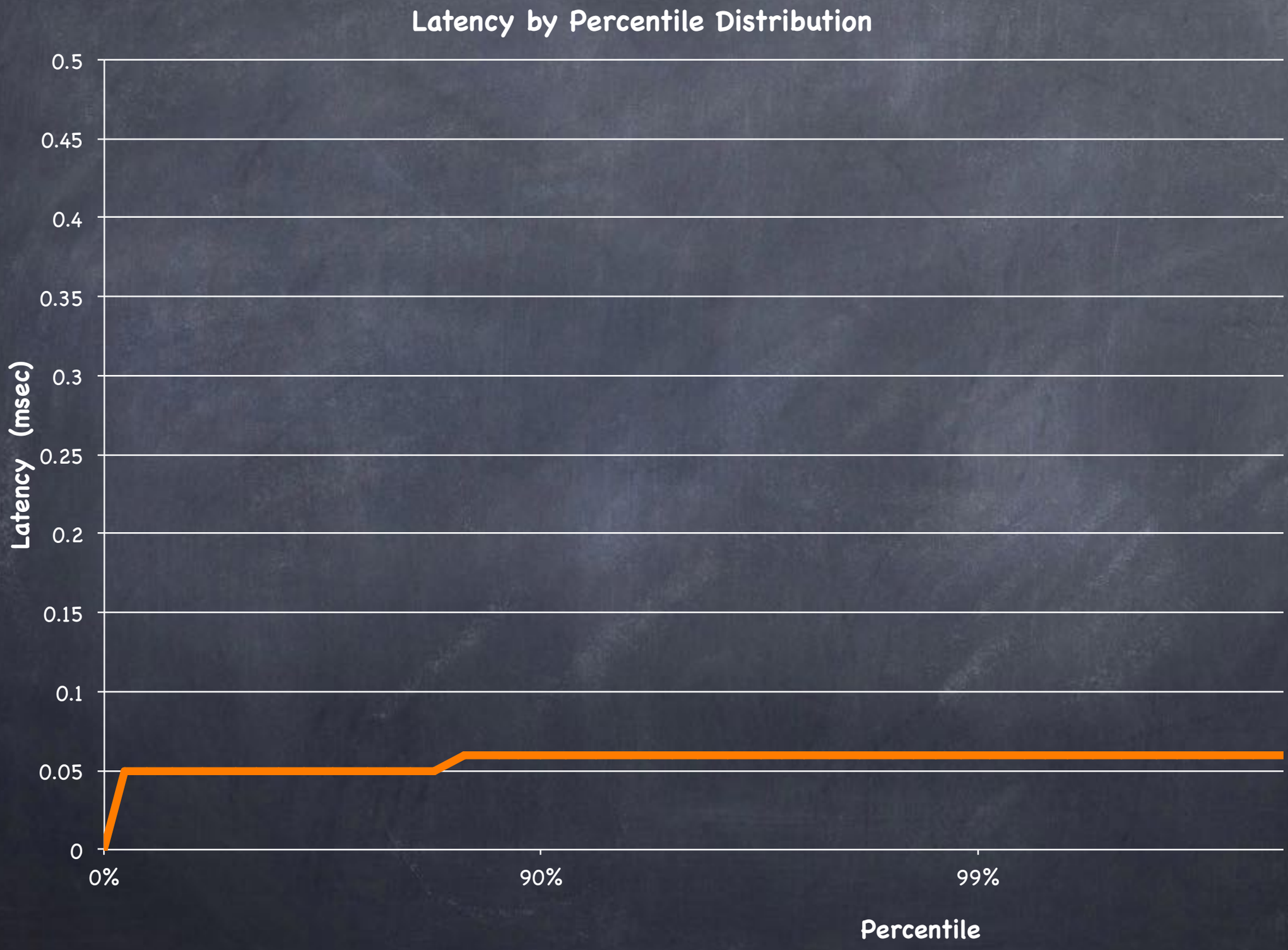
Latency behavior when bursts occur



200 usec service time, 5K msg/sec capacity
250 requests/sec, arriving in bursts of 50

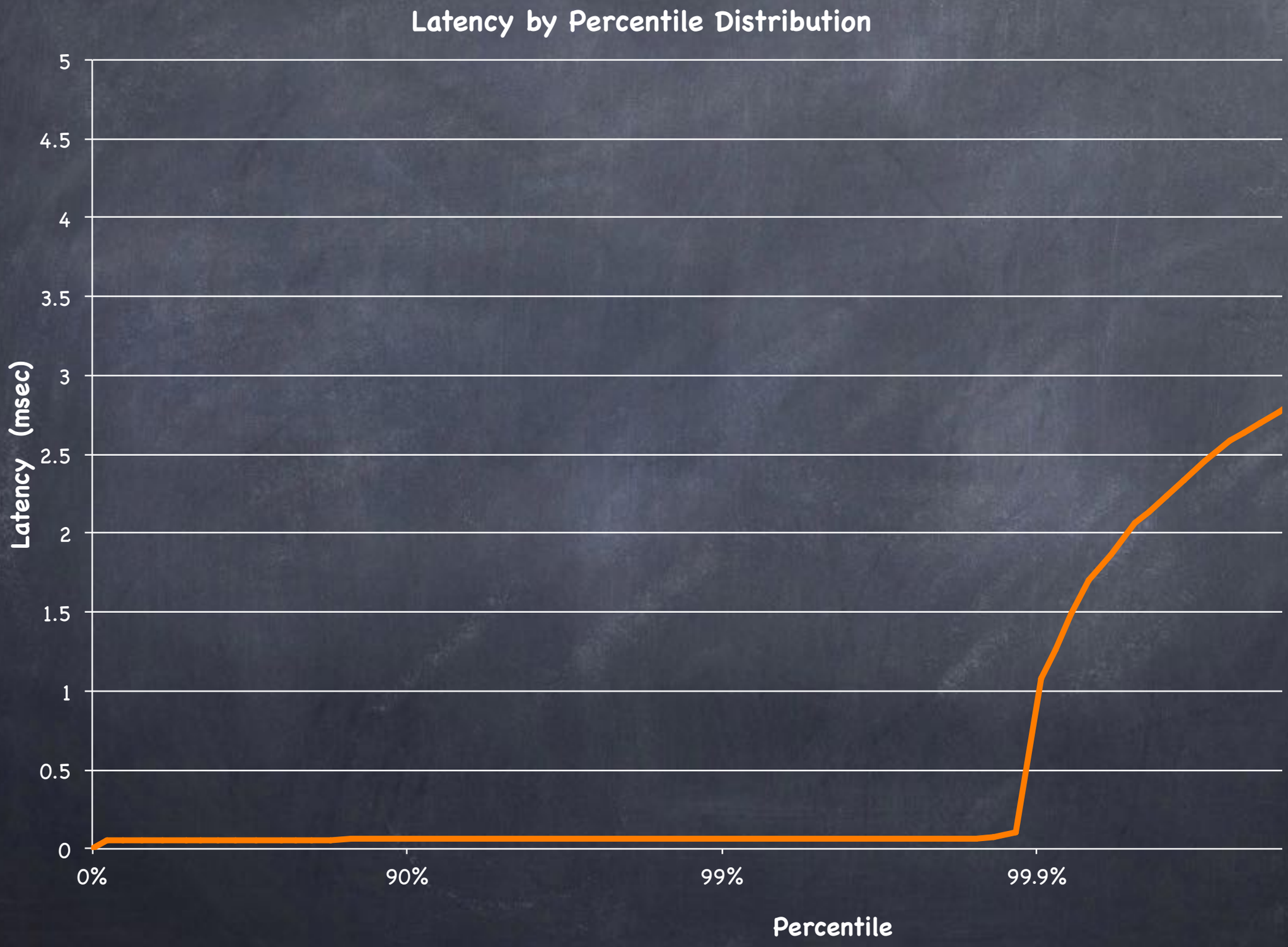
AZUL SYSTEMS

The real world: latency distribution



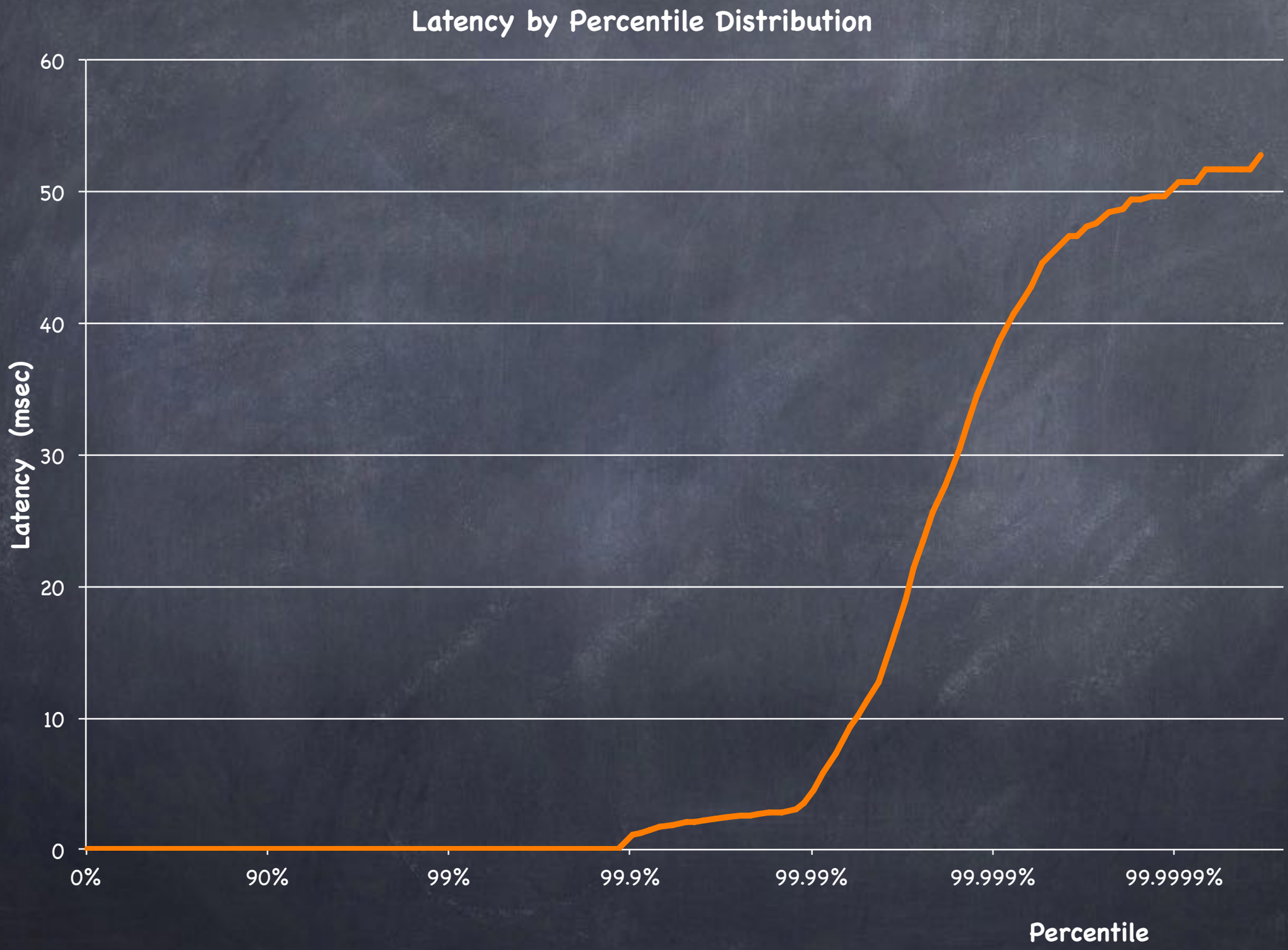
AZUL SYSTEMS

The real world: latency distribution



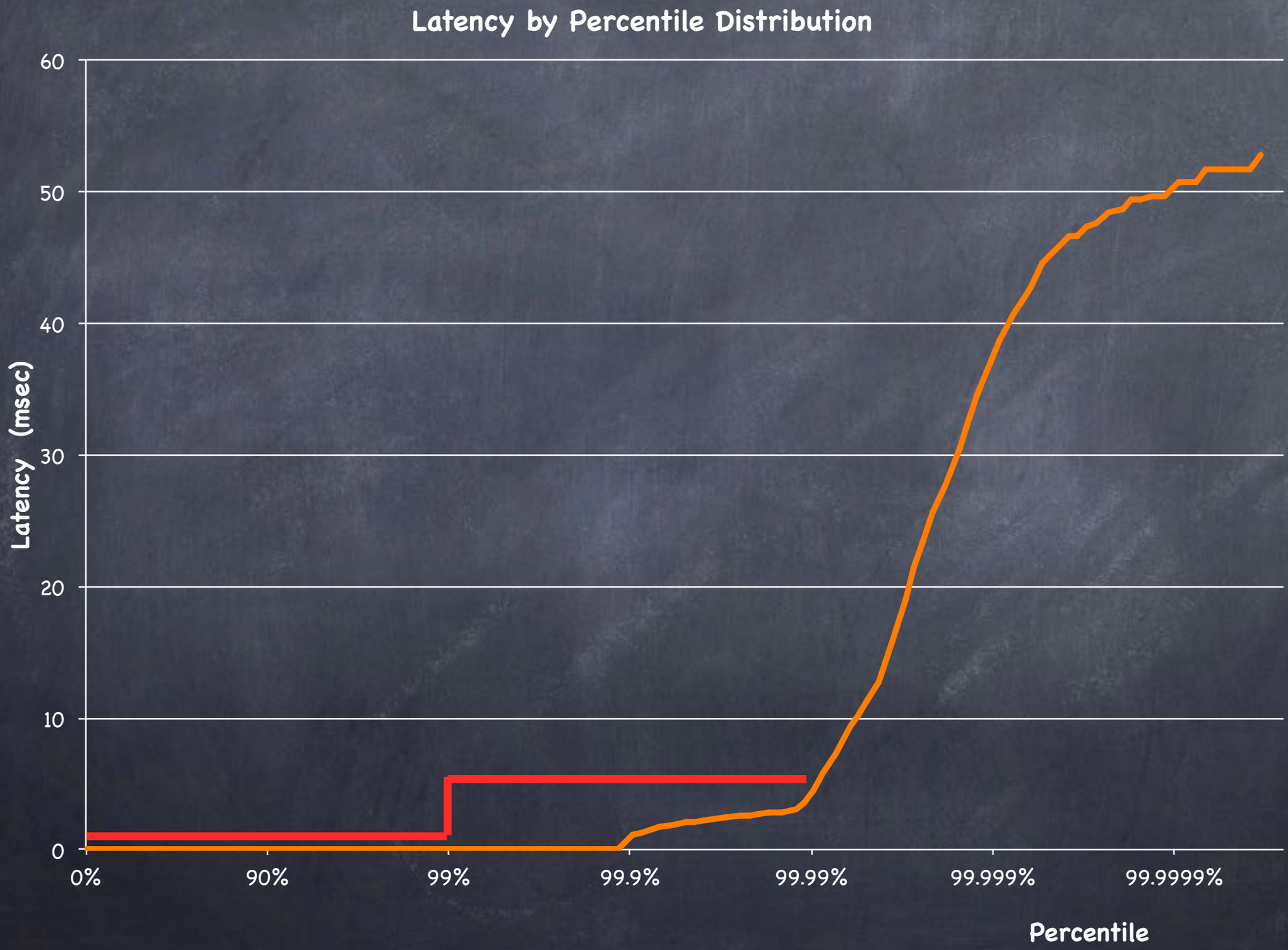
AZUL SYSTEMS

The real world: latency distribution



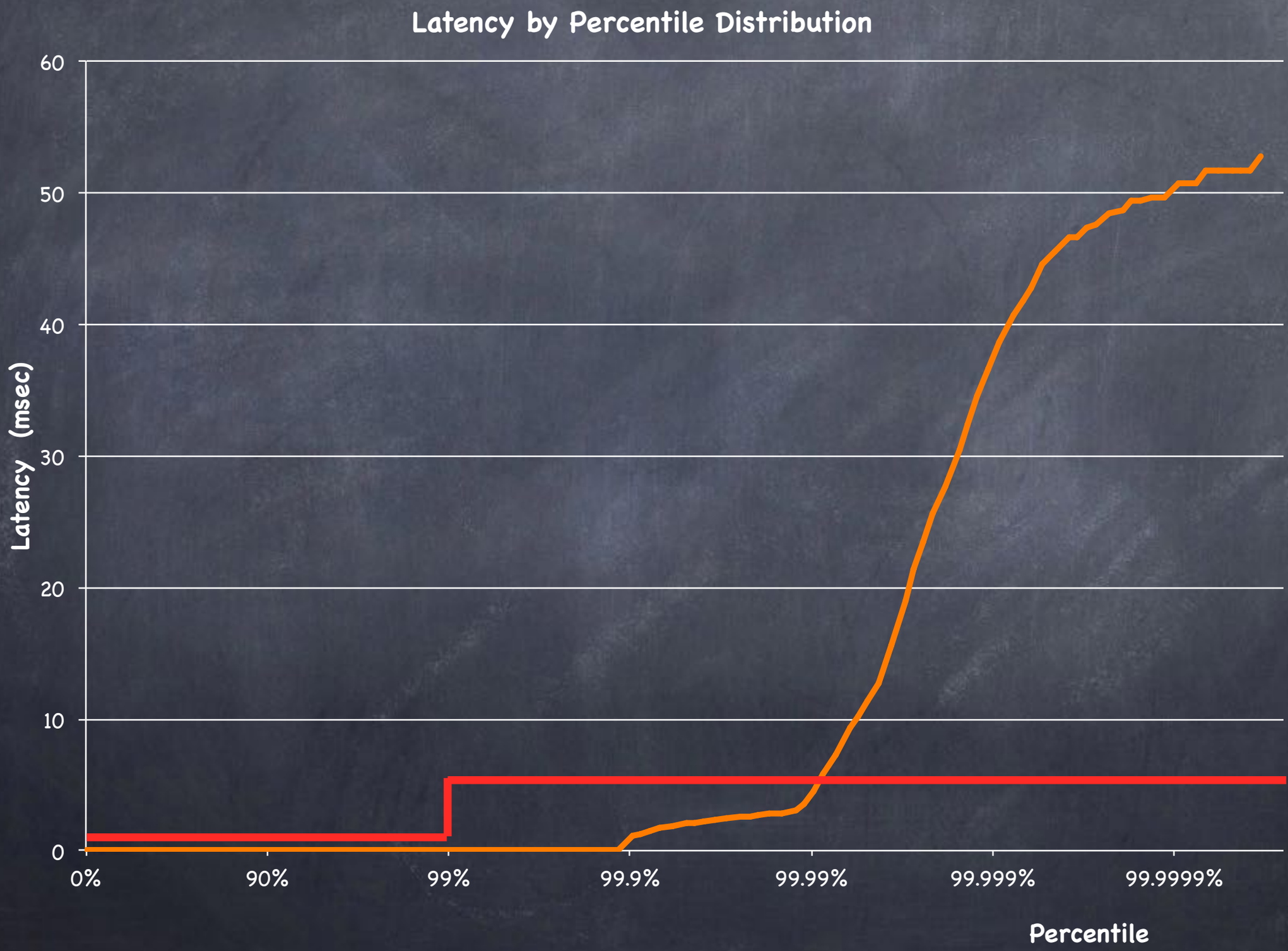
AZUL SYSTEMS

The real world: latency distribution



AZUL SYSTEMS

The real world: latency distribution



AZUL SYSTEMS

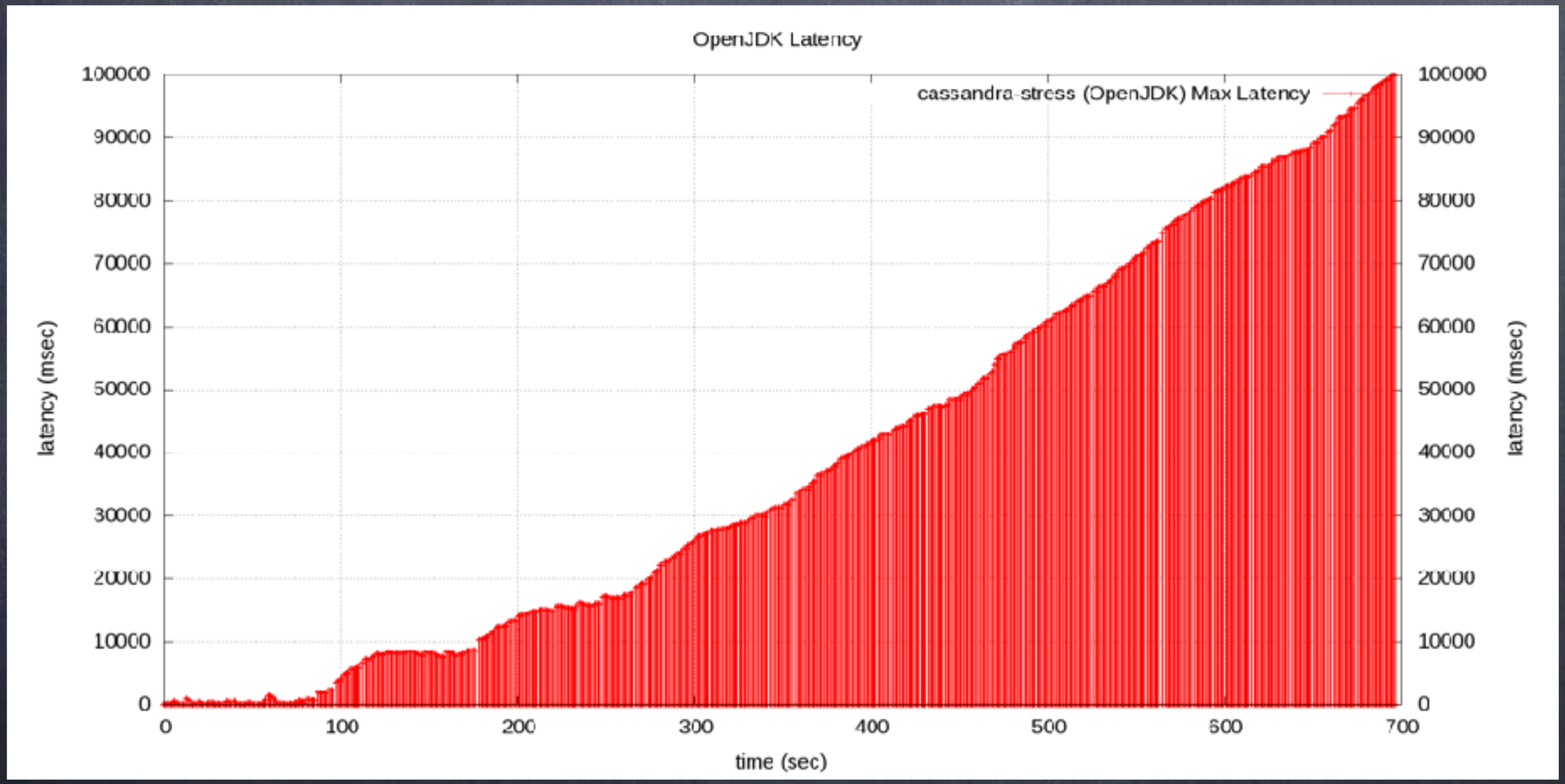
Service Time vs. Response Time

AZUL SYSTEMS

Lines can get pretty long…

AZUL SYSTEMS

Response time grows when incoming rate is consistently faster than what we can handle



AZUL SYSTEMS

Cutting corners in performance testing

AZUL SYSTEMS

Typical Reaction

AZUL SYSTEMS

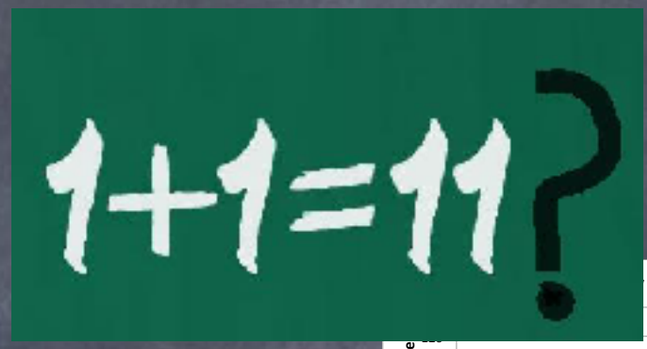
And most commonly

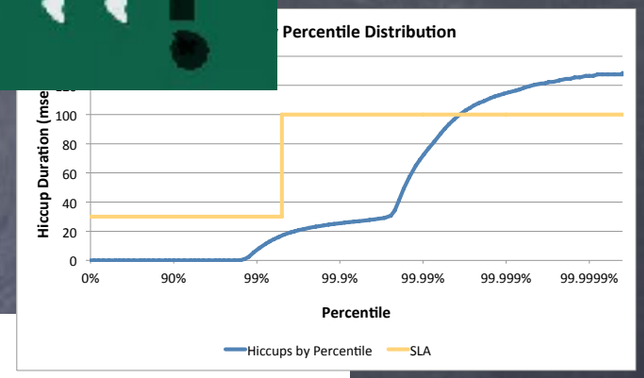
Repeat. Repeat. Repeat.

AZUL SYSTEMS

So before you start to measure something like widgets/sec:

- Establish requirements





  - Correctness
  - Timeliness
  - Availability, etc.
- Understand expected environmental limitations
  - Governing bottlenecks and realities

AZUL SYSTEMS

And most importantly

DO NOT consider "performance results"
from non-passing tests

AZUL SYSTEMS

Managed Runtimes

What's so special about them?

AZUL SYSTEMS

?

?

Scala

?

Ruby

?

Java

Go

F#

Python

?

PHP

Clojure

C#

JavaScript

Erlang

?

Objective-C

?

C

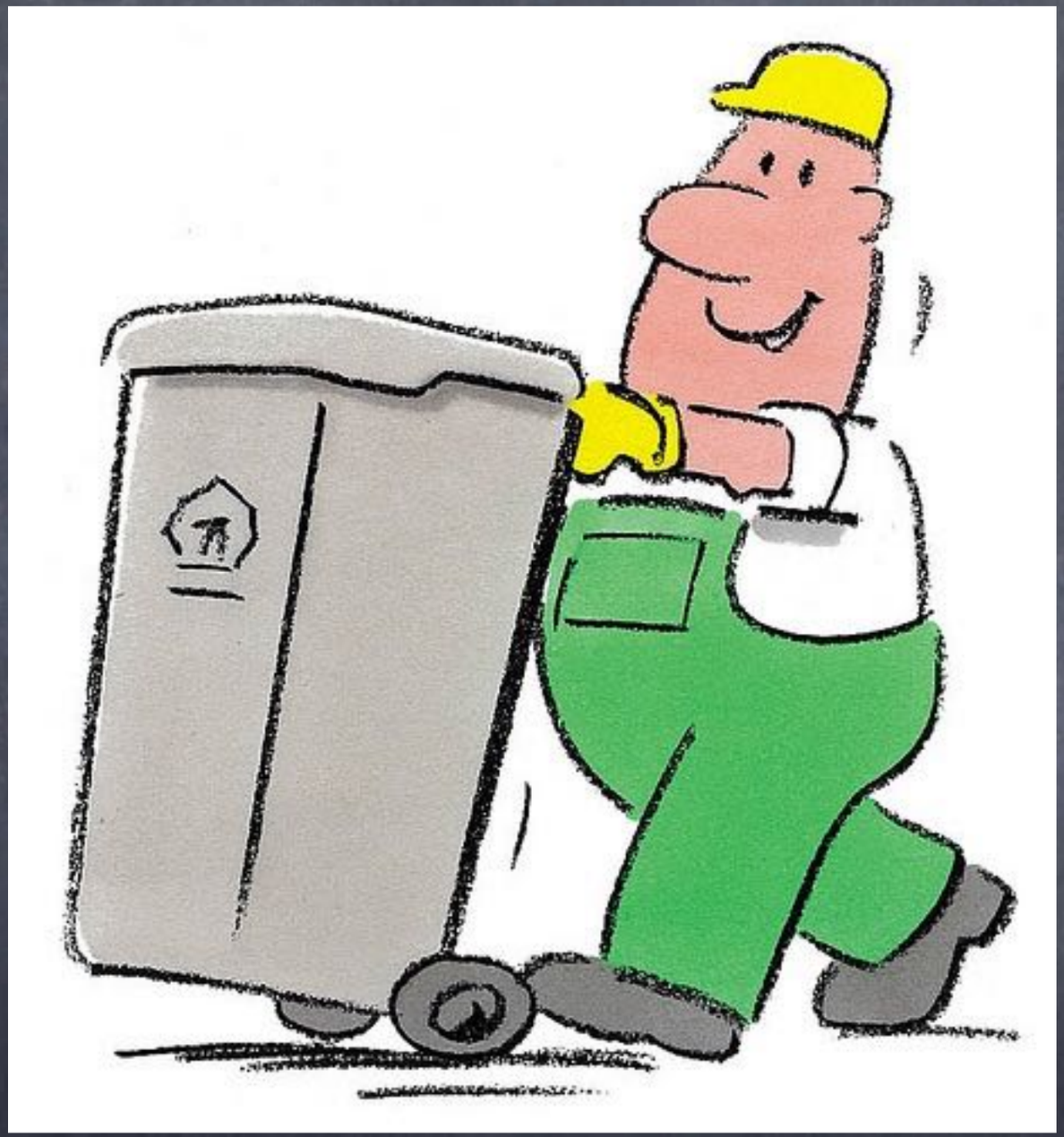
C++

?

?

?

AZUL SYSTEMS



AZUL SYSTEMS

Why?

AZUL SYSTEMS

Garbage Collection is… Good

- Productivity, stability
  - Programmers not responsible for freeing and destroying objects
  - Eliminates entire (common) areas of instability, delay, maintenance
- Guaranteed interoperability
  - No "memory management contract" needed across APIs
  - Uncoordinated libraries, frameworks, utilities seamlessly interoperate
- Facilitates practical use of large amounts of memory
  - Allows for complex and intertwined data structures
  - Within and across unrelated components

AZUL SYSTEMS

But most importantly

Garbage Collection makes things go fast faster

AZUL SYSTEMS

Time to market

AZUL SYSTEMS

Time to performance

AZUL SYSTEMS

Theoretical Performance

Fastest?

Fast

Slower?

AZUL SYSTEMS

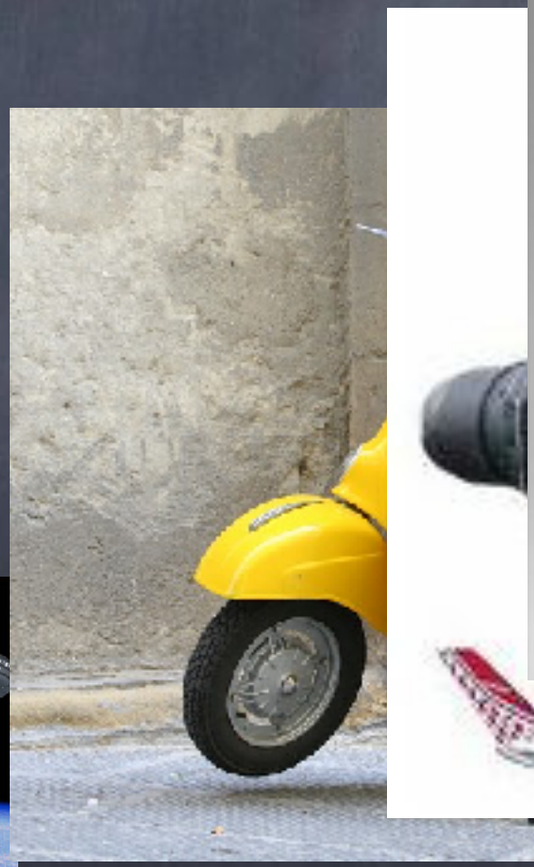
Pragmatic Performance

Which of these is fast enough
to get to work in 15 minutes or less?

AZUL SYSTEMS

Pragmatic Performance

Which will provide the needed speed
by [now + 6 months] ?

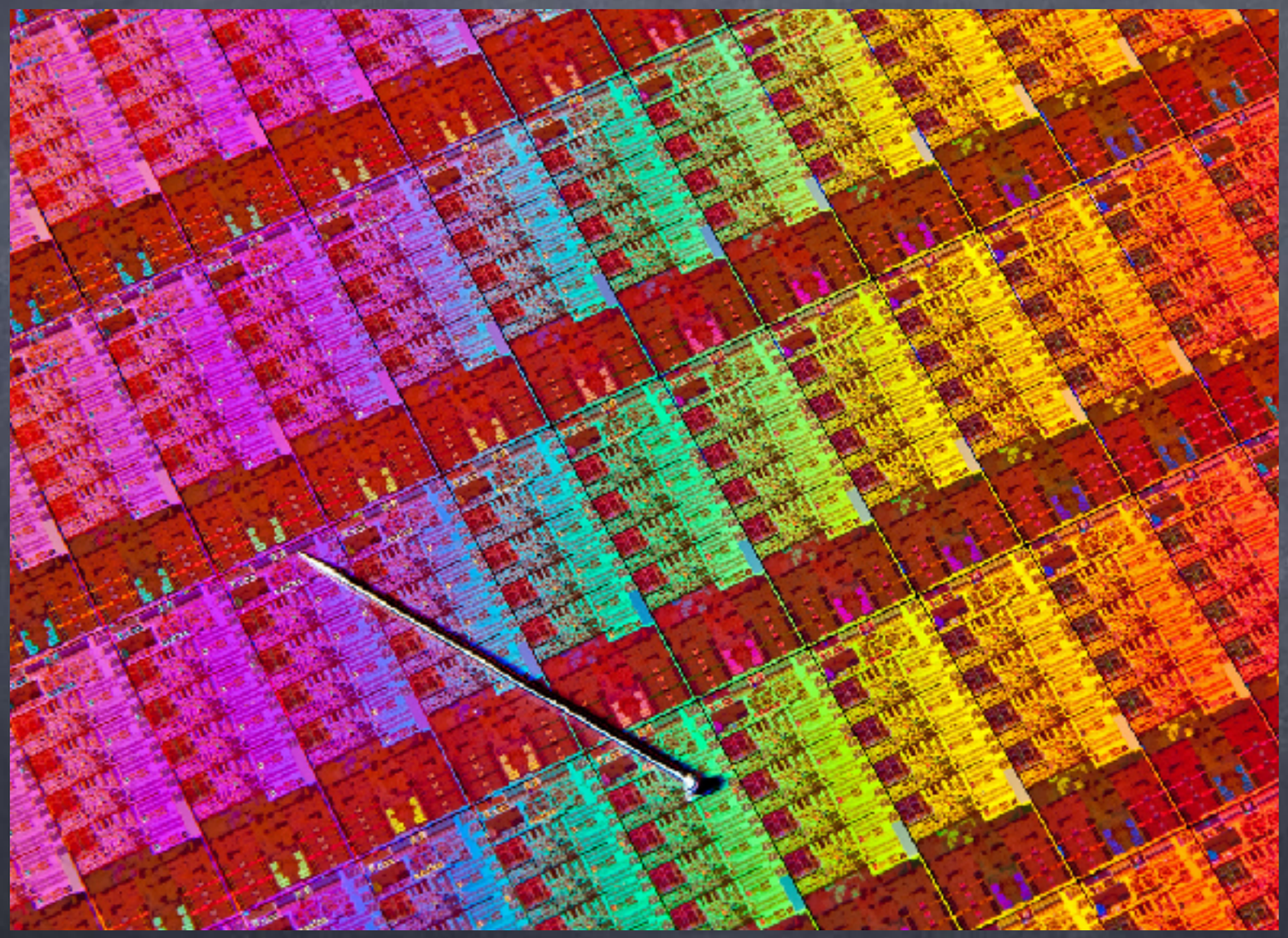


AZUL SYSTEMS

Tomorrow is here

AZUL SYSTEMS

Multicore is so 2012...



AZUL SYSTEMS

Current (2016) cloud stuff

Region: US East (N. Virginia)

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
Compute Optimized - Current Generation					
c4.large	2	8	3.75	EBS Only	$0.116 per Hour
c4.xlarge	4	16	7.5	EBS Only	$0.232 per Hour
c4.2xlarge	8	31	15	EBS Only	$0.464 per Hour
c4.4xlarge	16	62	30	EBS Only	$0.928 per Hour
c4.8xlarge	36	132	60	EBS Only	$1.856 per Hour
Memory Optimized - Current Generation					
x1.16xlarge	64	174.5	976	1 x 1920 SSD	$6.669 per Hour
x1.32xlarge	128	349	1952	2 x 1920 SSD	$13.338 per Hour
r3.large	2	6.5	15	1 x 32 SSD	$0.166 per Hour
r3.xlarge	4	13	30.5	1 x 80 SSD	$0.333 per Hour
r3.2xlarge	8	26	61	1 x 160 SSD	$0.665 per Hour
r3.4xlarge	16	52	122	1 x 320 SSD	$1.33 per Hour
r3.8xlarge	32	104	244	2 x 320 SSD	$2.66 per Hour

AZUL SYSTEMS

"lots" of cores

"lots" of memory

"Waste"

and

Performance

AZUL SYSTEMS

AZUL SYSTEMS

"Waste"

AZUL SYSTEMS

@giltene

Summary

- Pragmatic Performance
  - What actually needs to get done?
  - How much. How quickly. How fast.
  - What needs to remain true while we do the stuff
- Performance is (usually) not about efficiency
  - unless efficiency is your performance metric
- Do not be afraid to come up with creative "waste"

AZUL SYSTEMS

Q & A

@giltene

http://www.azulsystems.com