How NOT to Measure Latency

Gil Tene, CTO & co-Founder, Azul Systems
@giltene

AZUL SYSTEMS

The "Oh S@%#!" talk

Gil Tene, CTO & co-Founder, Azul Systems
@giltene

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 also depress people by pulling the wool up from over their eyes...

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

AZUL SYSTEMS

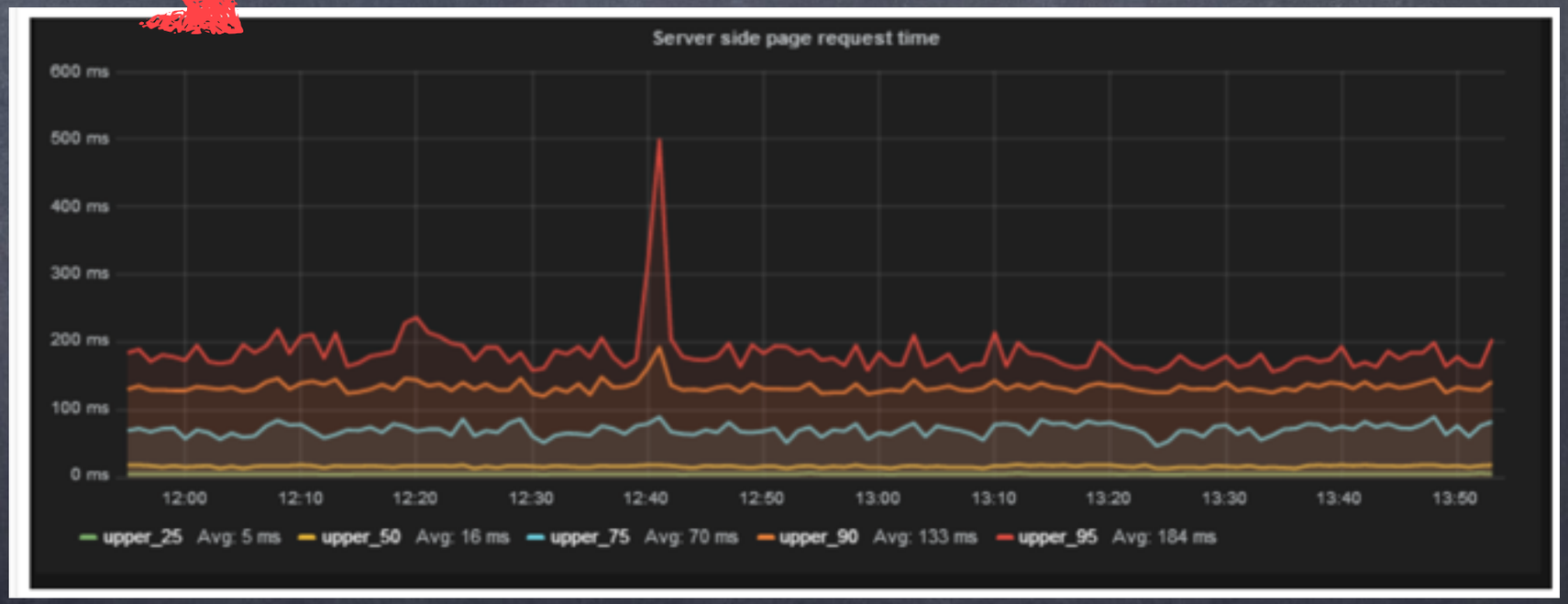
Latency Behavior

- Latency: The time it took one operation to happen
- Each operation occurrence has its own latency
- What we care about is how latency behaves
- Behavior is a lot more than "the common case was X"

AZUL SYSTEMS

We like to look at pretty charts…

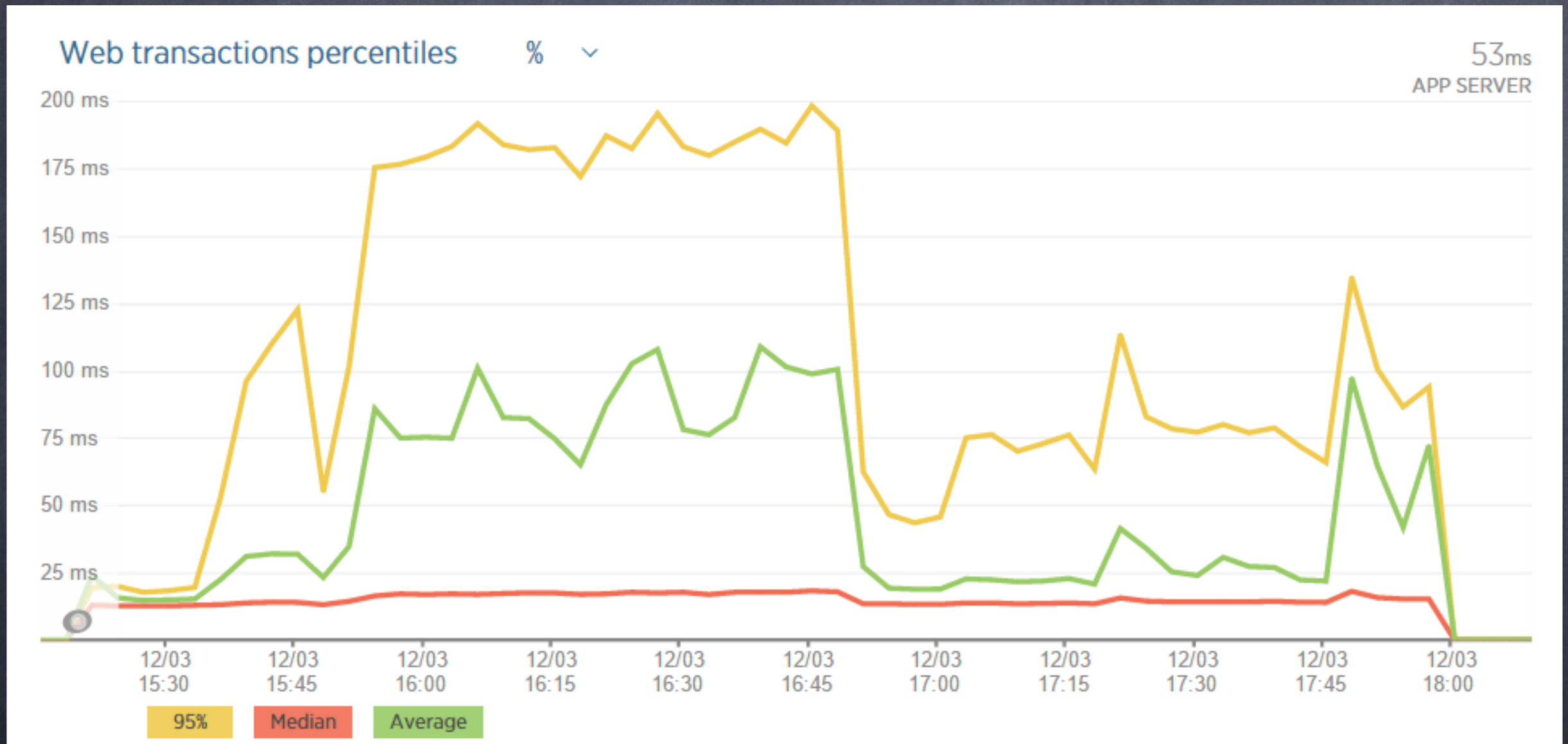
95%'lie



The "We only want to show good things" chart

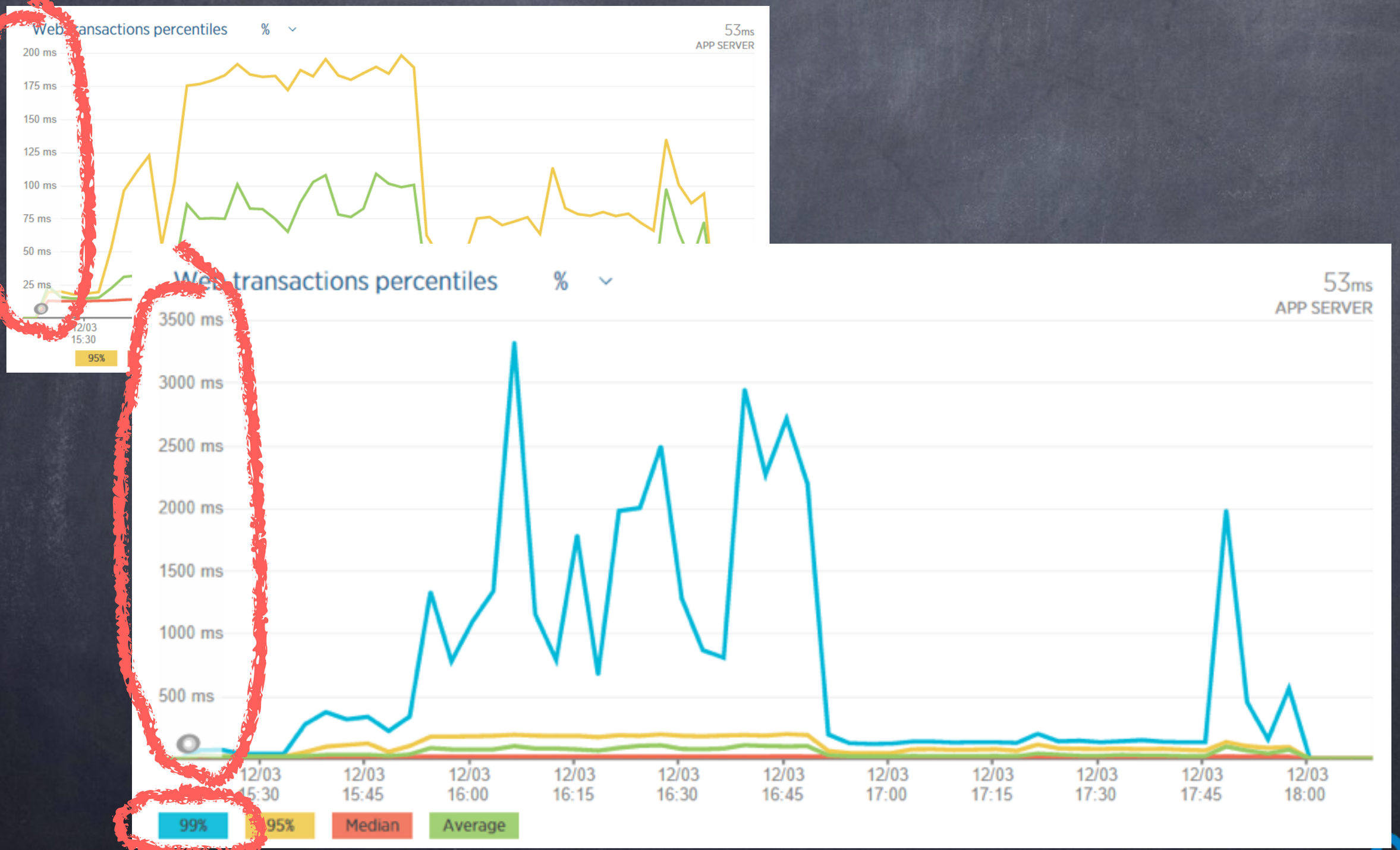
AZUL SYSTEMS

A real world, real time example



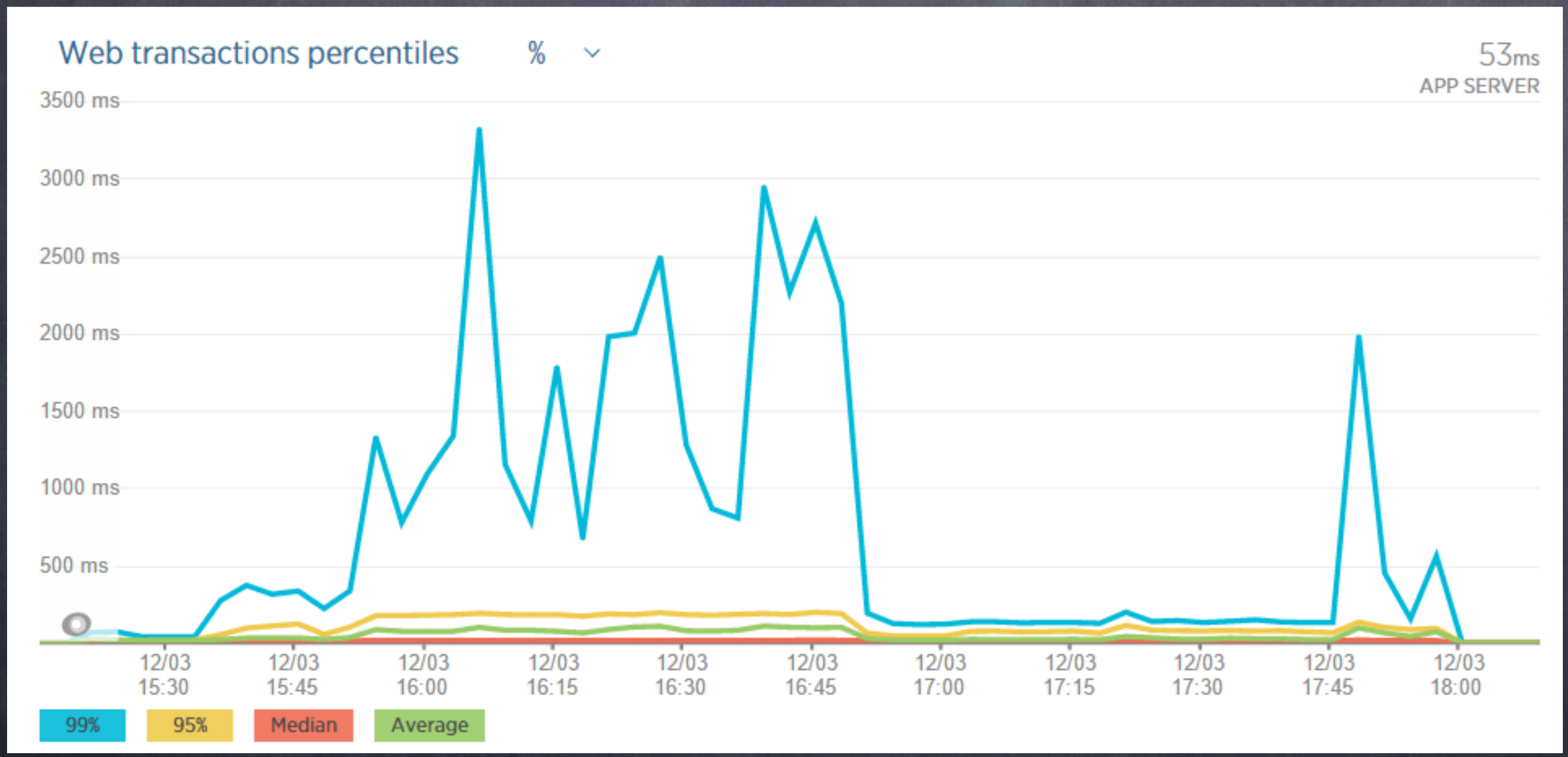
AZUL SYSTEMS

A real world, real time example



AZUL SYSTEMS

A real world, real time example



So this is a better picture. Right?

AZUL SYSTEMS

I like to rant about latency…

About Me

Gil Tene

CTO and co-founder of Azul Systems.

View my complete profile

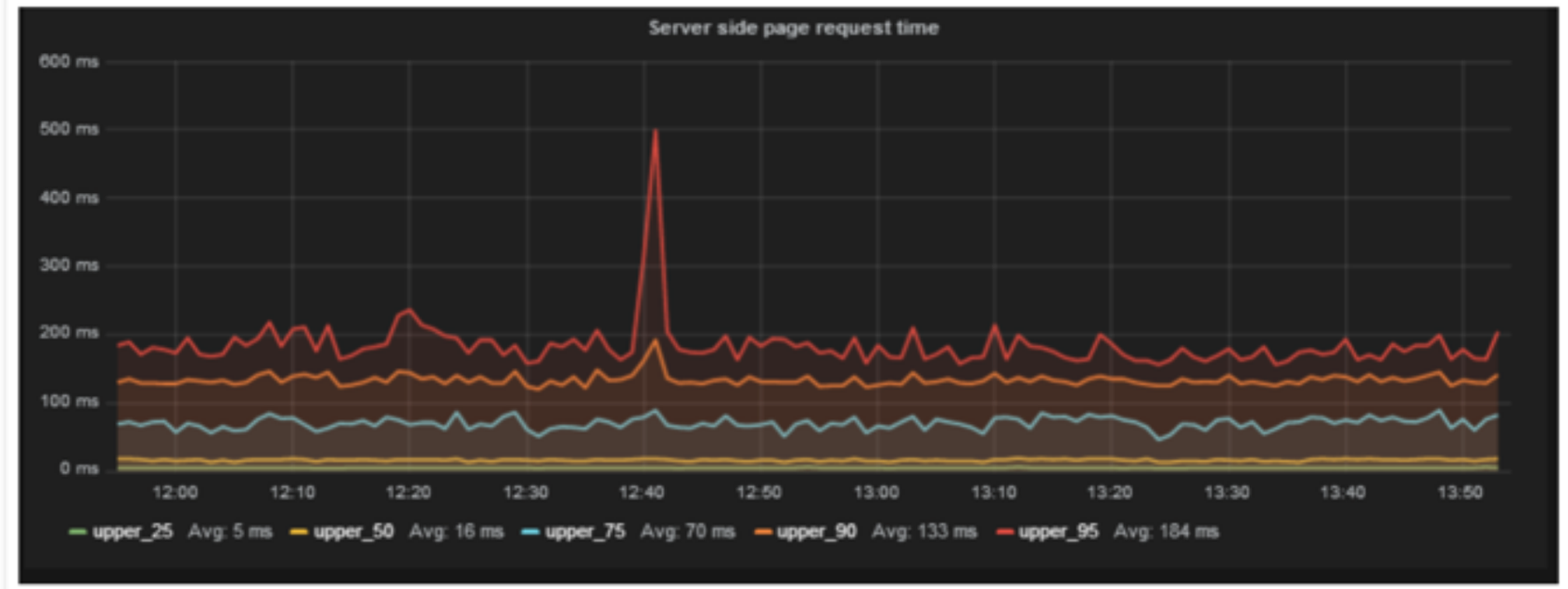
Blog Archive

- 2014 (8)
  - June (8)
    - #LatencyTipOfTheDay: Median Server Response Time: ...
    - #LatencyTipOfTheDay: MOST page loads will experien...
    - #LatencyTipOfTheDay: Q: What's wrong with this pic...
    - #LatencyTipOfTheDay: If you are not measuring and/...
    - #LatencyTipOfTheDay : Measure what you need to mon...
    - #LatencyTipOfTheDay: Average (def): a random numbe...

Saturday, June 21, 2014

#LatencyTipOfTheDay: Q: What's wrong with this picture? A: Everything!

Question: What's wrong with this picture:



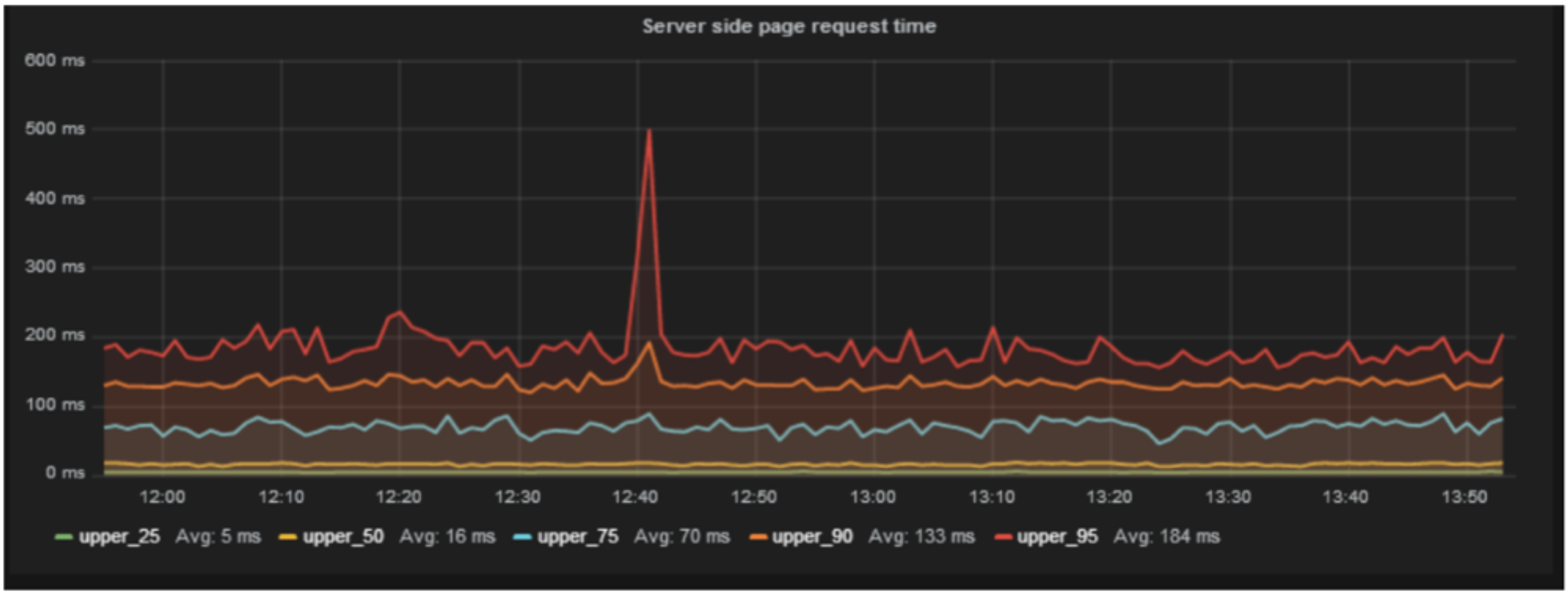
Answer: Everything!

AZUL SYSTEMS

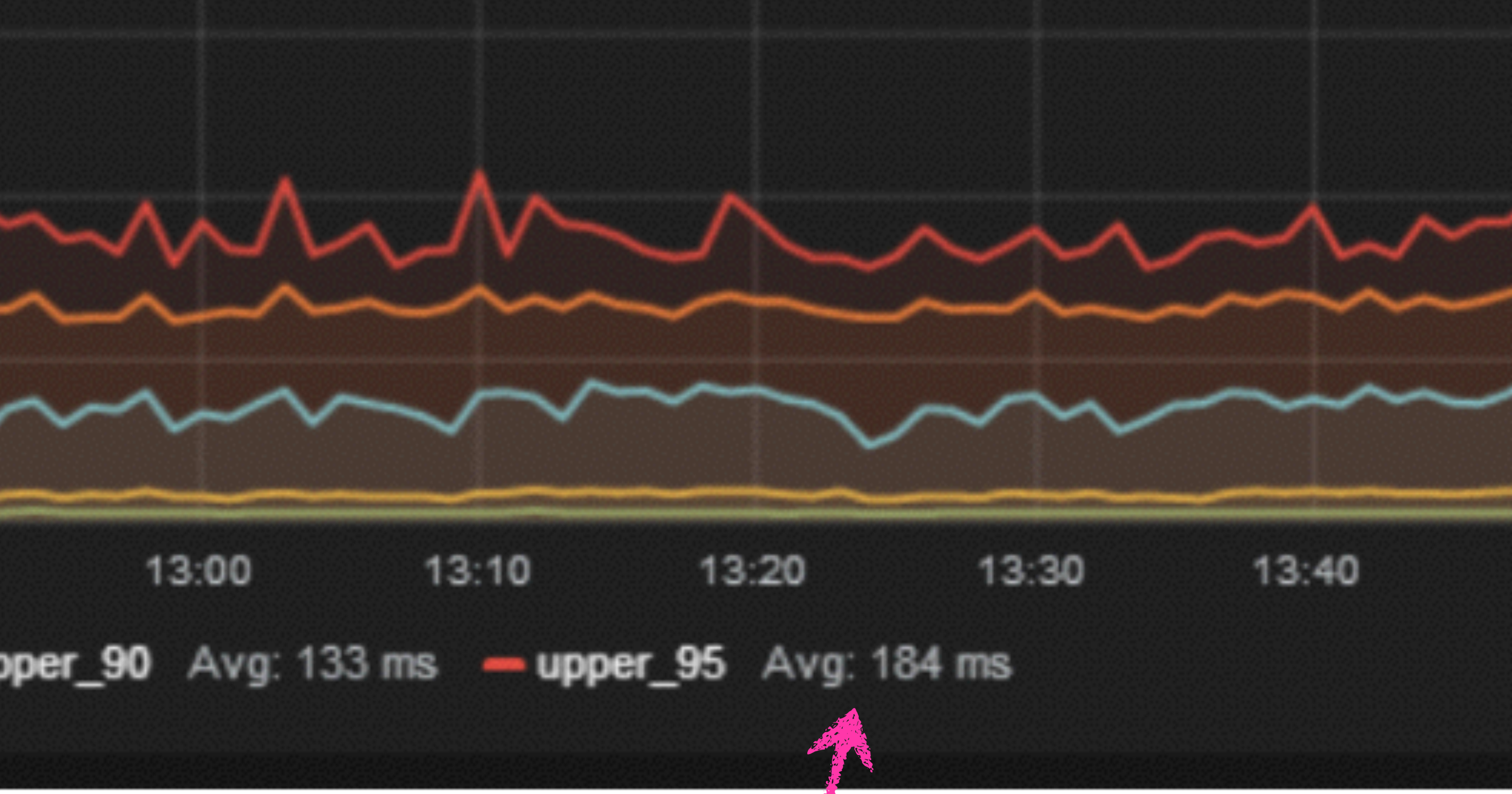
#LatencyTipOfTheDay:

If you are not measuring and/or plotting Max, what are you hiding (from)?

AZUL SYSTEMS



AZUL SYSTEMS



What (TF) does the Average of the 95%'lie mean?

AZUL SYSTEMS

What (TF) does the Average of the 95%'lie mean?

- Lets do the same with 100%'ile; Suppose we a set of 100%'ile values for each minute:

[1, 0, 3, 1, 601, 4, 2, 8, 0, 3, 3, 1, 1, 0, 2]

"The average 100%'ile over the past 15 minutes was 42"

- Same nonsense applies to any other %'lie

AZUL SYSTEMS

#LatencyTipOfTheDay:

You can't average percentiles.

Period.

AZUL SYSTEMS

Percentiles Matter

AZUL SYSTEMS

Is the 99%'lie "rare"?

AZUL SYSTEMS

99%'lie: a good indicator, right?

What are the chances of a single web page view experiencing >99%'lie latency of:

- A single search engine node?

- A single Key/Value store node?

- A single Database node?

- A single CDN request?

AZUL SYSTEMS

Site	# of requests
amazon.com	190
kohls.com	204
jcrew.com	112
saksfifthavenue.com	109
--	--
nytimes.com	173
cnn.com	279
--	--
twitter.com	87
pinterest.com	84
facebook.com	178
--	--
google.com (yes, that simple noise-free page)	31
google.com search for "http requests per page"	76

AZUL SYSTEMS

Site	# of requests	page loads that would experience the 99%'lie [(1 - (.99 ^ N)) * 100%]
amazon.com	190	85.2%
kohls.com	204	87.1%
jcrew.com	112	67.6%
saksfifthavenue.com	109	66.5%
--	--	--
nytimes.com	173	82.4%
cnn.com	279	93.9%
--	--	--
twitter.com	87	58.3%
pinterest.com	84	57.0%
facebook.com	178	83.3%
--	--	--
google.com (yes, that simple noise-free page)	31	26.7%
google.com search for "http requests per page"	76	53.4%

AZUL SYSTEMS

#LatencyTipOfTheDay:

MOST page loads will experience the 99%'lie server response

AZUL SYSTEMS

Which HTTP response time metric is more "representative" of user experience?

The 95%'lie or the 99.9%'lie

AZUL SYSTEMS

Gauging user experience

Example: If a typical user session involves 5 page loads, averaging 40 resources per page.

- How many of our users will NOT experience something worse than the 95%'lie of http requests?

Answer: ~0.003%

- How may of our users will experience at least one response that is longer than the 99.9%'lie?

Answer: ~18%

AZUL SYSTEMS

Gauging user experience

Example: If a typical user session involves 5 page loads, averaging 40 resources per page.

- What http response percentile will be experienced by the 95%'ile of users?

Answer: ~99.97%

- What http response percentile will be experienced by the 99%'ile of users

Answer: ~99.995%

AZUL SYSTEMS

#LatencyTipOfTheDay:

Median Server Response Time:
The number that 99.9999999999%
of page views can be worse than

AZUL SYSTEMS

Why don’t we have response time or latency stats with multiple 9s in them???

AZUL SYSTEMS

Why don’t we have response time or latency stats with multiple 9s in them???

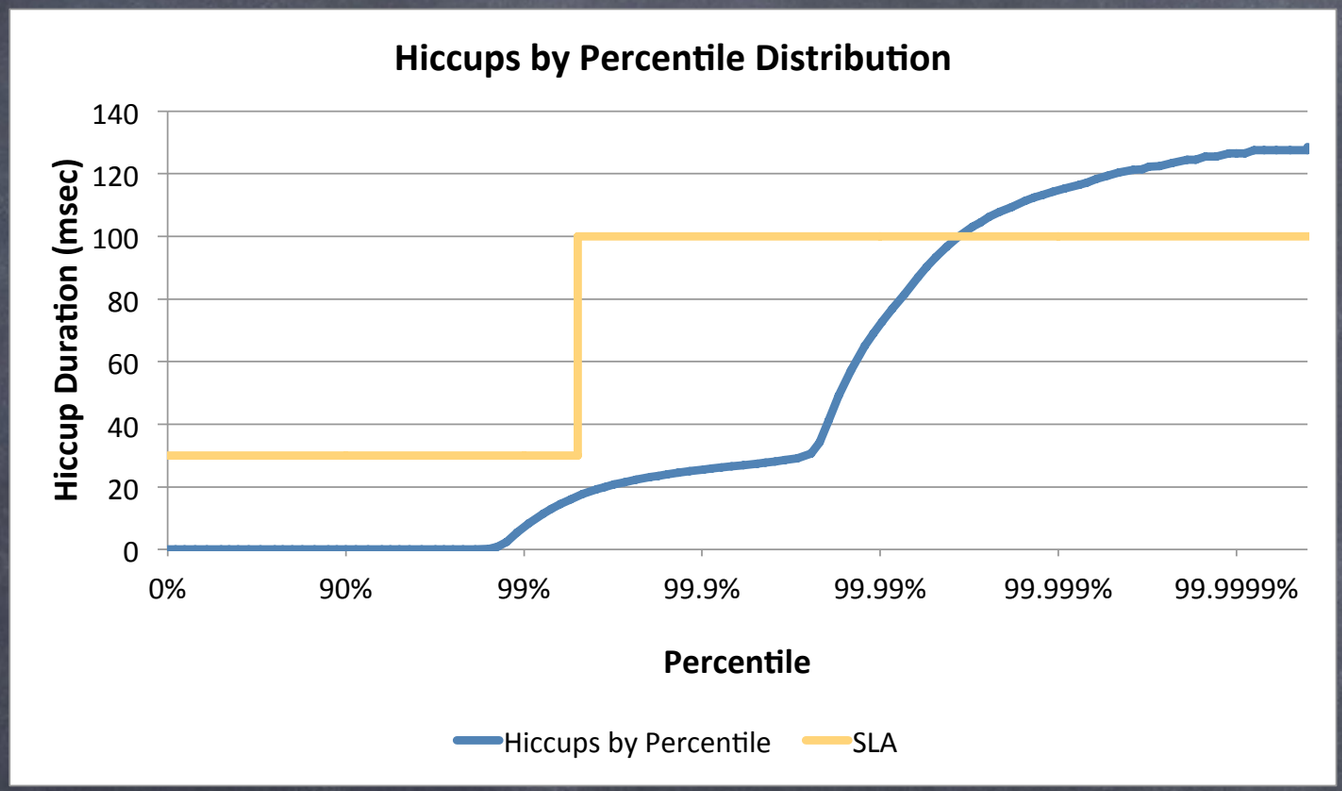
You can’t average percentiles…

And you also can’t get an hour’s 99.999%’lie out of lots of 10 second interval 99%’lie reports…

AZUL SYSTEMS

Why don’t we have response time or latency stats with multiple 9s in them???

Check out HdrHistogram



You can’t average percentiles…

It lets you have nice things….

And you also can’t get an hour’s 99.999%’lie out of lots of 10 second interval 99%’lie reports…

AZUL SYSTEMS

The coordinated omission problem

An accidental conspiracy...

The *lie* in the 99%'lies

AZUL SYSTEMS

The coordinated omission problem

- Common Example A (load testing):
  - each "client" issues requests at a certain rate
  - measure/log response time for each request

- So what's wrong with that?
  - works **only** if ALL responses fit within interval
  - implicit "automatic back off" coordination

AZUL SYSTEMS

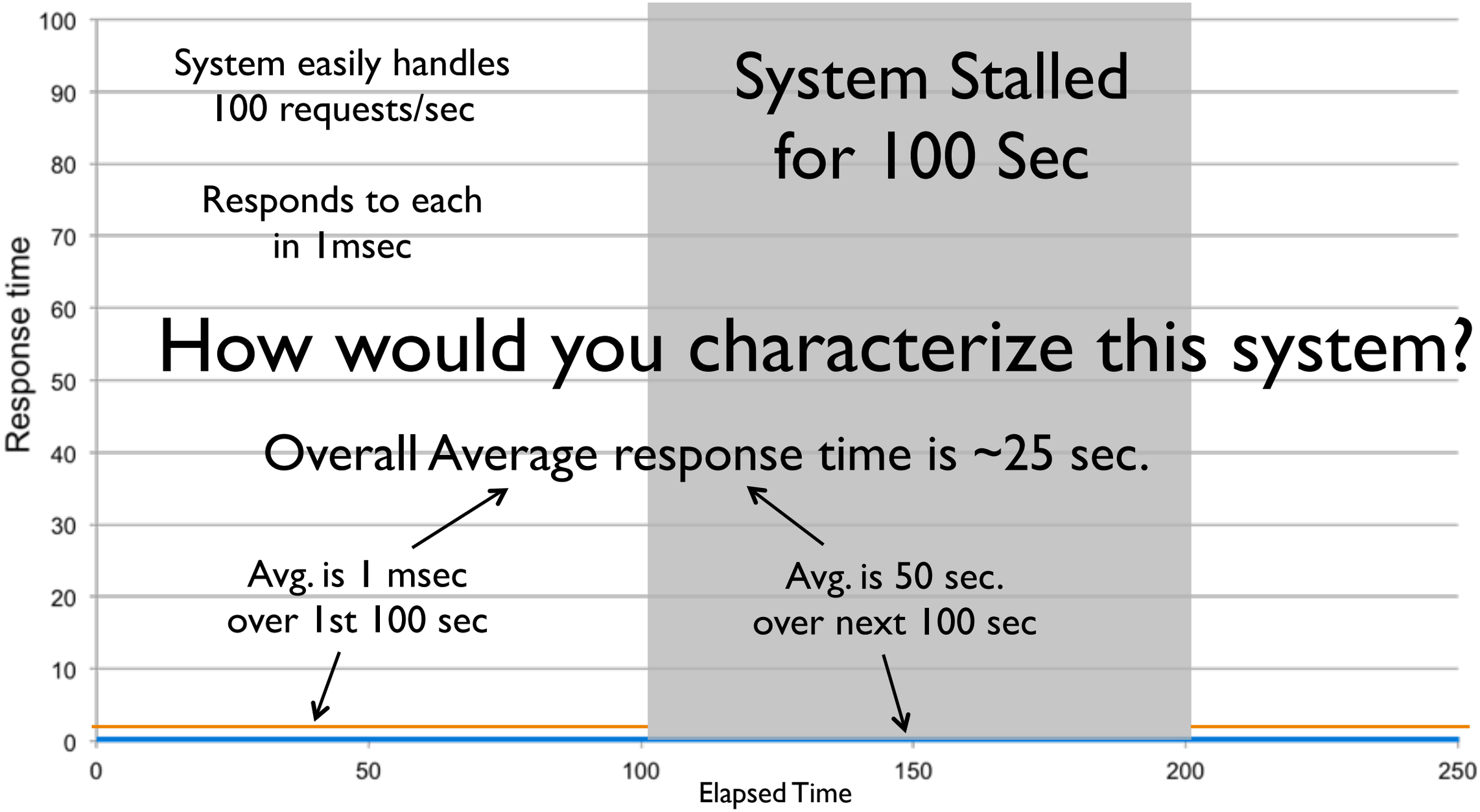
Common Example B: Coordinated Omission in Monitoring Code

```
/**
 * Performs the actual reading of a row out of the StorageService, fetching
 * a specific set of column names from a given column family.
 */
public static List<Row> read(List<ReadCommand> commands, ConsistencyLevel consistency_level)
        throws UnavailableException, IsBootstrappingException, ReadTimeoutException
{
    if (StorageService.instance.isBootstrapMode())
        throw new IsBootstrappingException();
    long startTime = System.nanoTime();
    List<Row> rows;
    try
    {
        rows = fetchRows(commands, consistency_level);
    }
    finally
    {
        readMetrics.addNano(System.nanoTime() - startTime);
    }
    return rows;
}
```

- Long operations only get measured once
- delays outside of timing window do not get measured at all

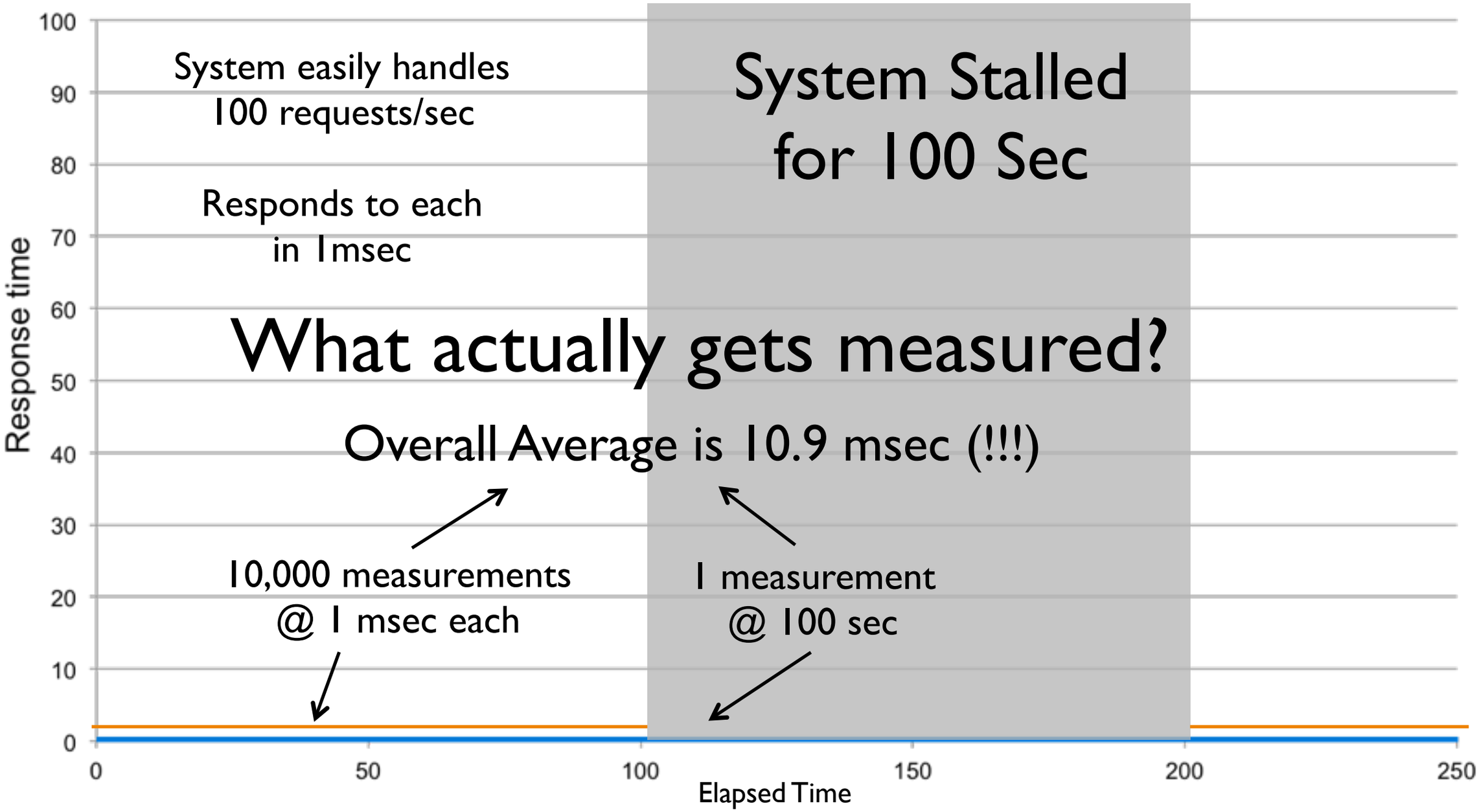
AZUL SYSTEMS

How bad can this get?



~50%‘ile is 1 msec ~75%‘ile is 50 sec 99.99%‘ile is ~100sec

Measurement in practice



50%'ile is 1 msec

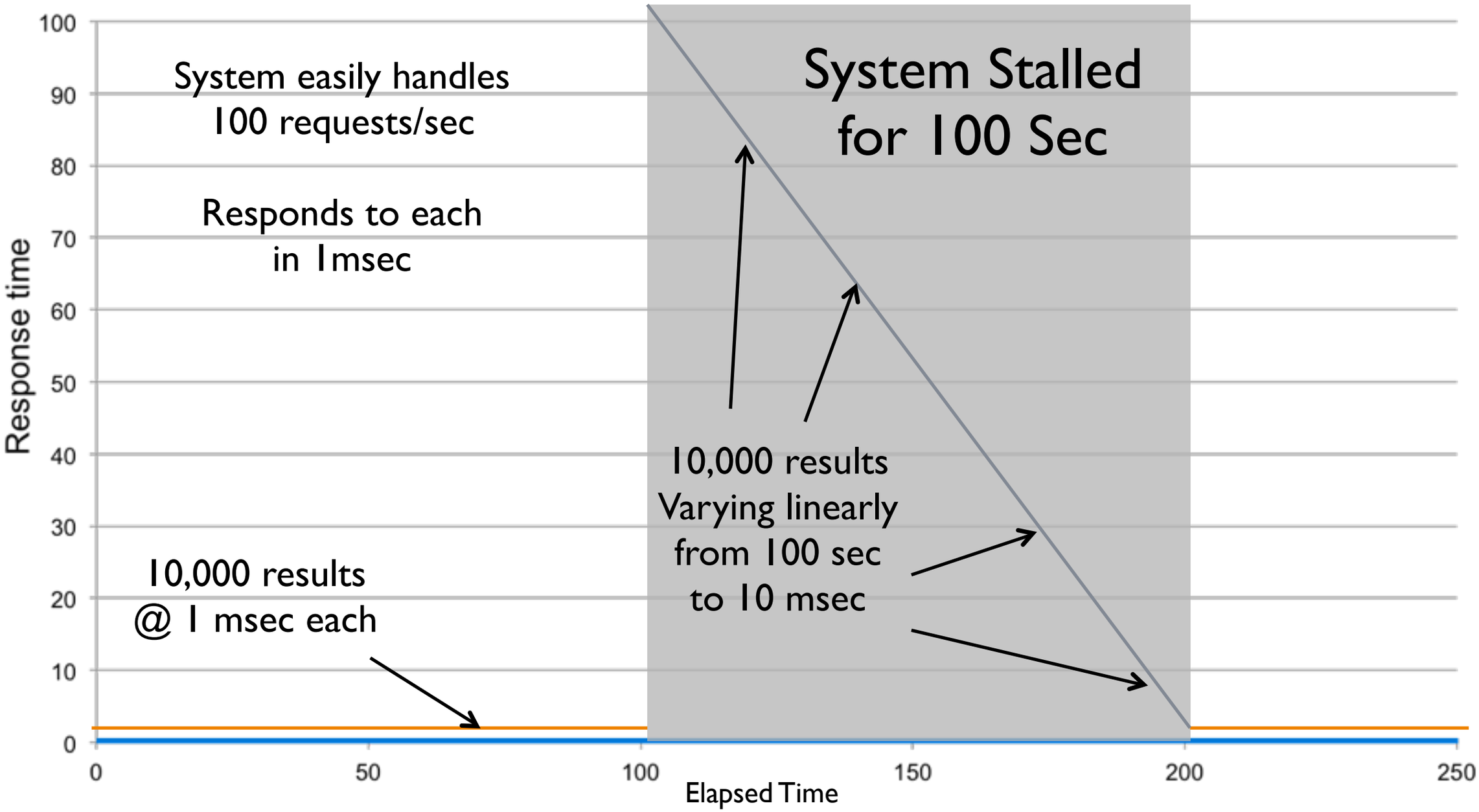
75%'lie is 1 msec

(should be ~50sec)

99.99%'lie is 1 msec

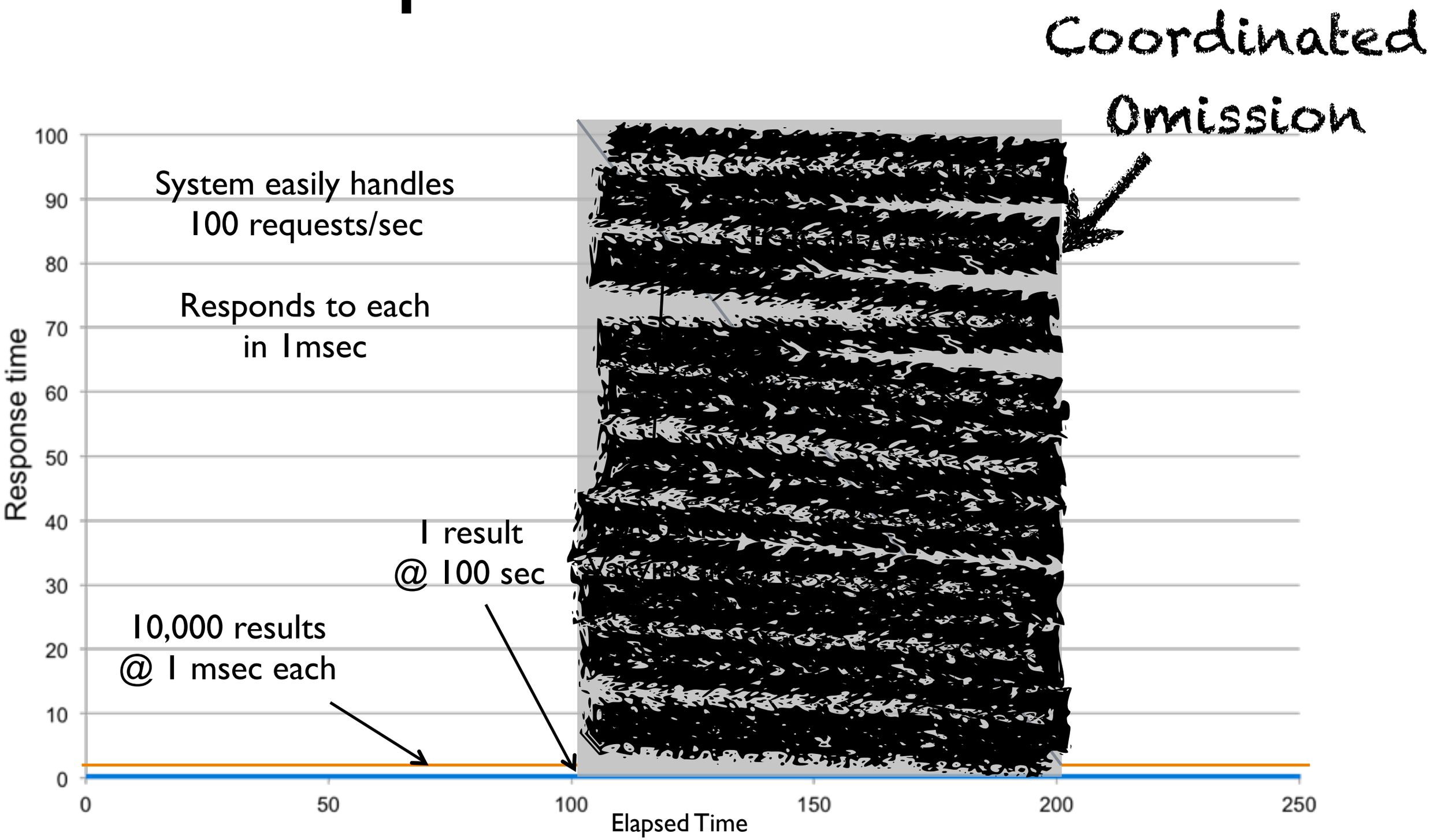
(should be ~100 sec)

Proper measurement



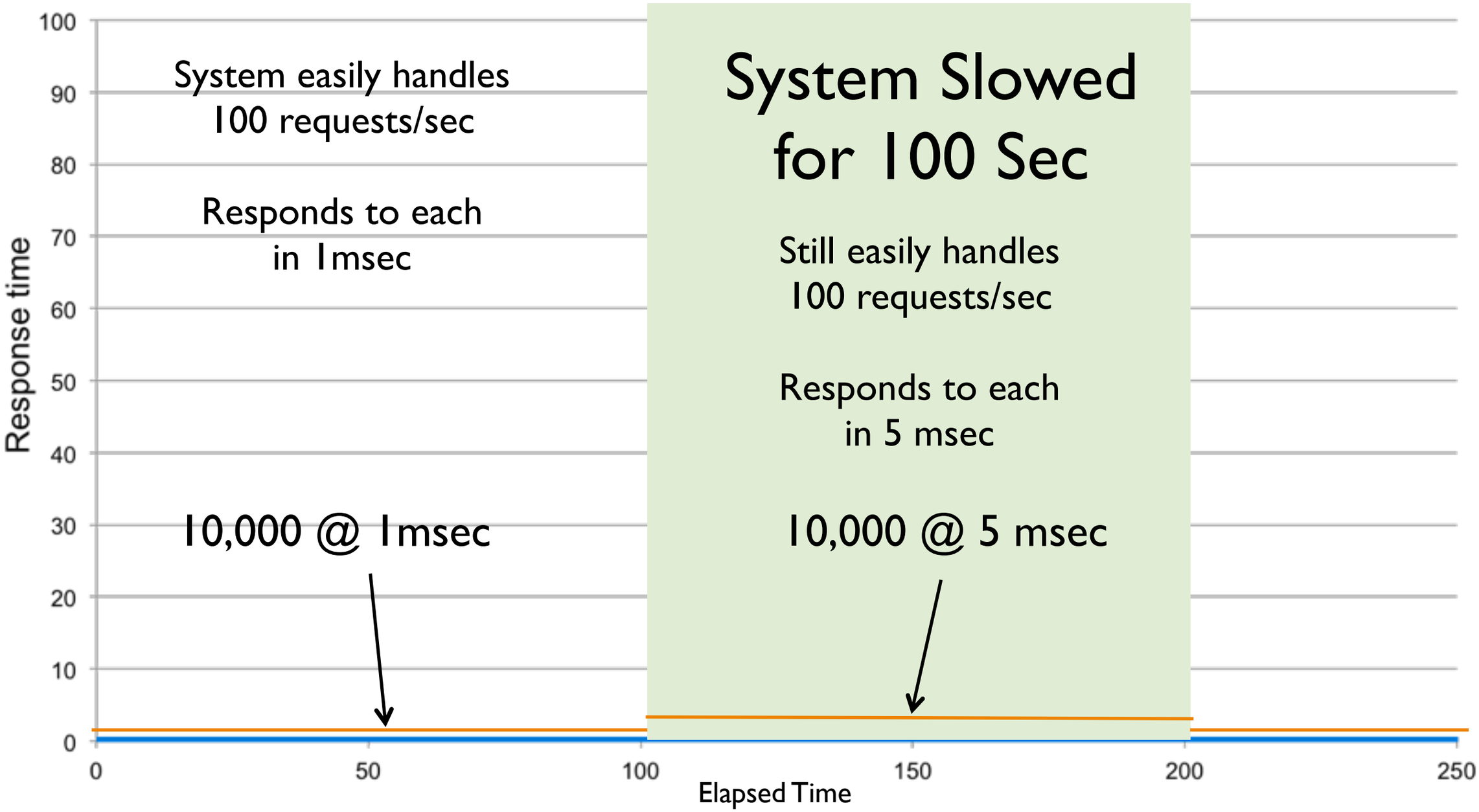
~50%‘ile is 1 msec ~75%‘ile is 50 sec 99.99%‘ile is ~100sec

Proper measurement



~50%‘ile is 1 msec ~75%‘ile is ~~50 sec~~ 1 msec 99.99%‘ile is ~~100 sec~~ 1 msec

"Better" can look "Worse"

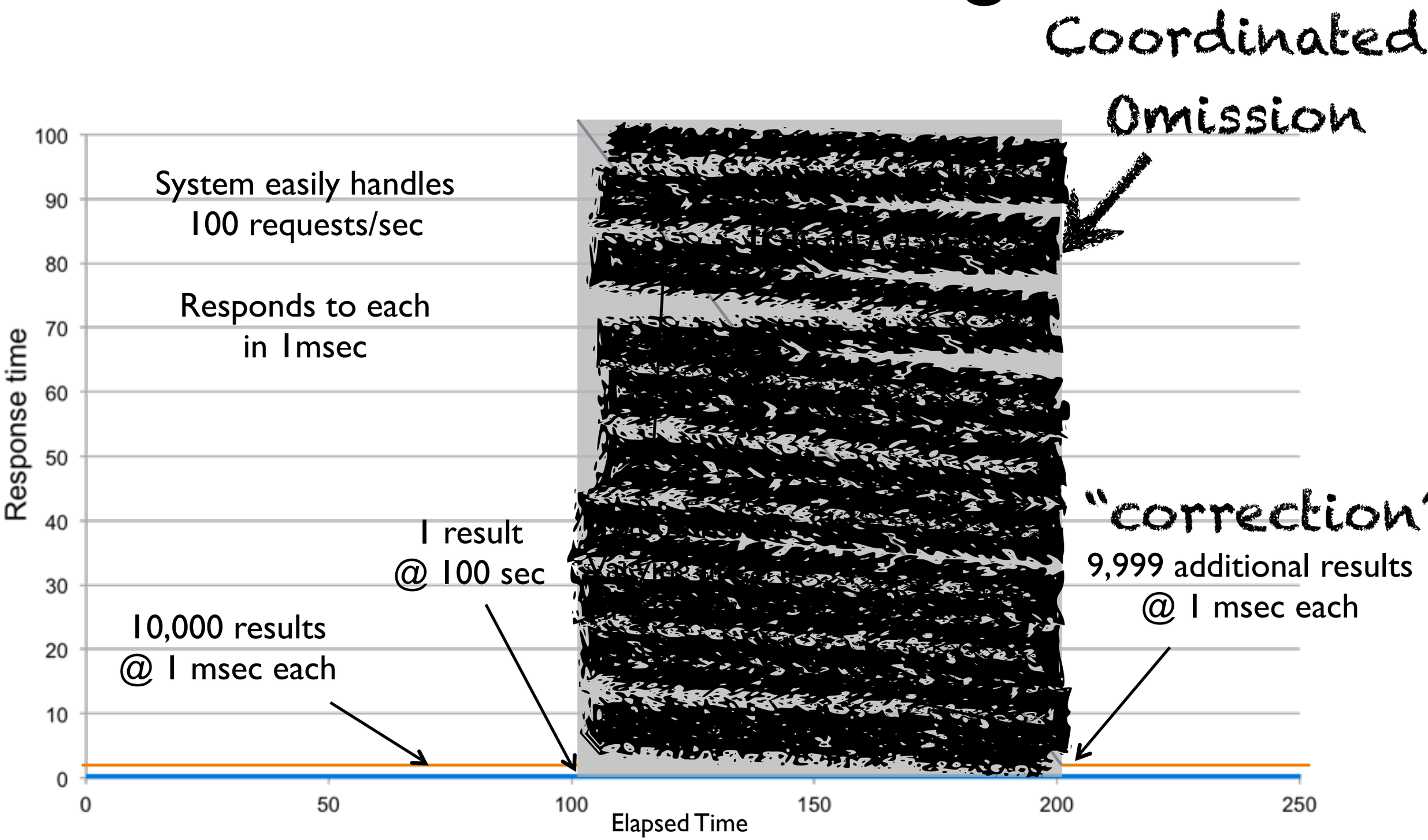


50%'ile is 1 msec

75%'lie is 2.5msec
(stalled shows 1 msec)

99.99%'lie is ~5msec
(stalled shows 1 msec)

"Correction": "Cheating Twice"



~50%'ile is 1 msec

~75%'ile is ~~50 sec~~ 1 msec

99.994%'ile is ~~100 sec~~ 1 msec

Response Time vs. Service Time

AZUL SYSTEMS

Service Time vs. Response Time

AZUL SYSTEMS

Coordinated Omission

Usually

makes something that you ***think*** is a

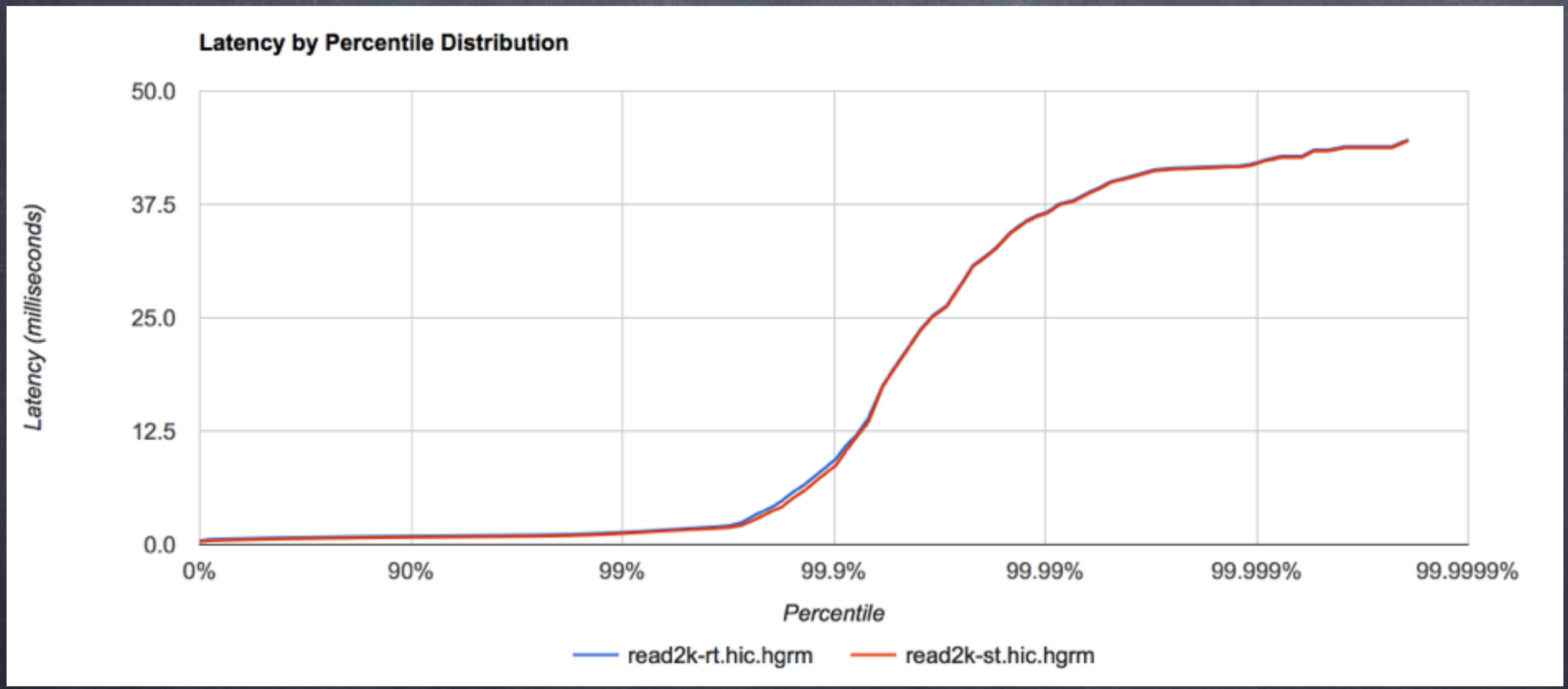
Response Time metric

only represent

the ***Service Time*** component

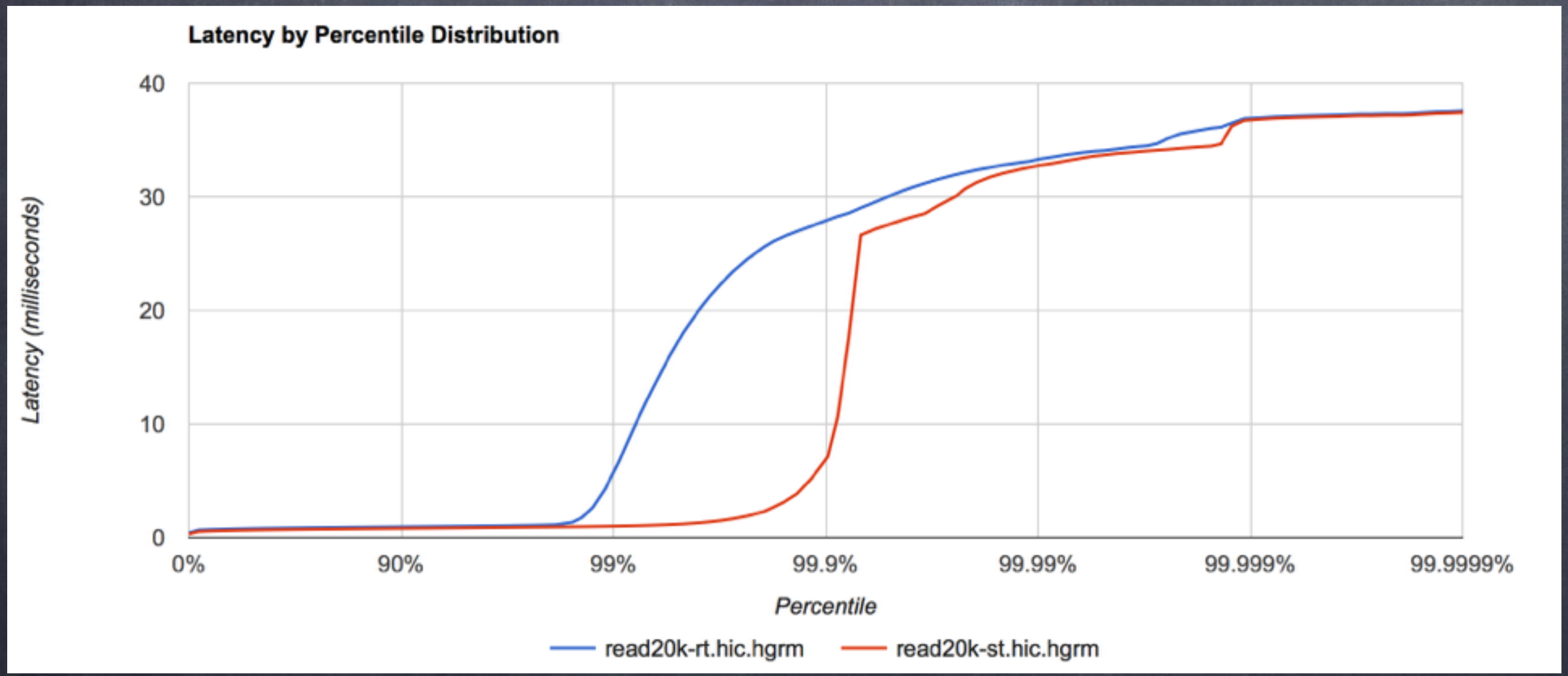
AZUL SYSTEMS

Response Time vs. Service Time @2K/sec



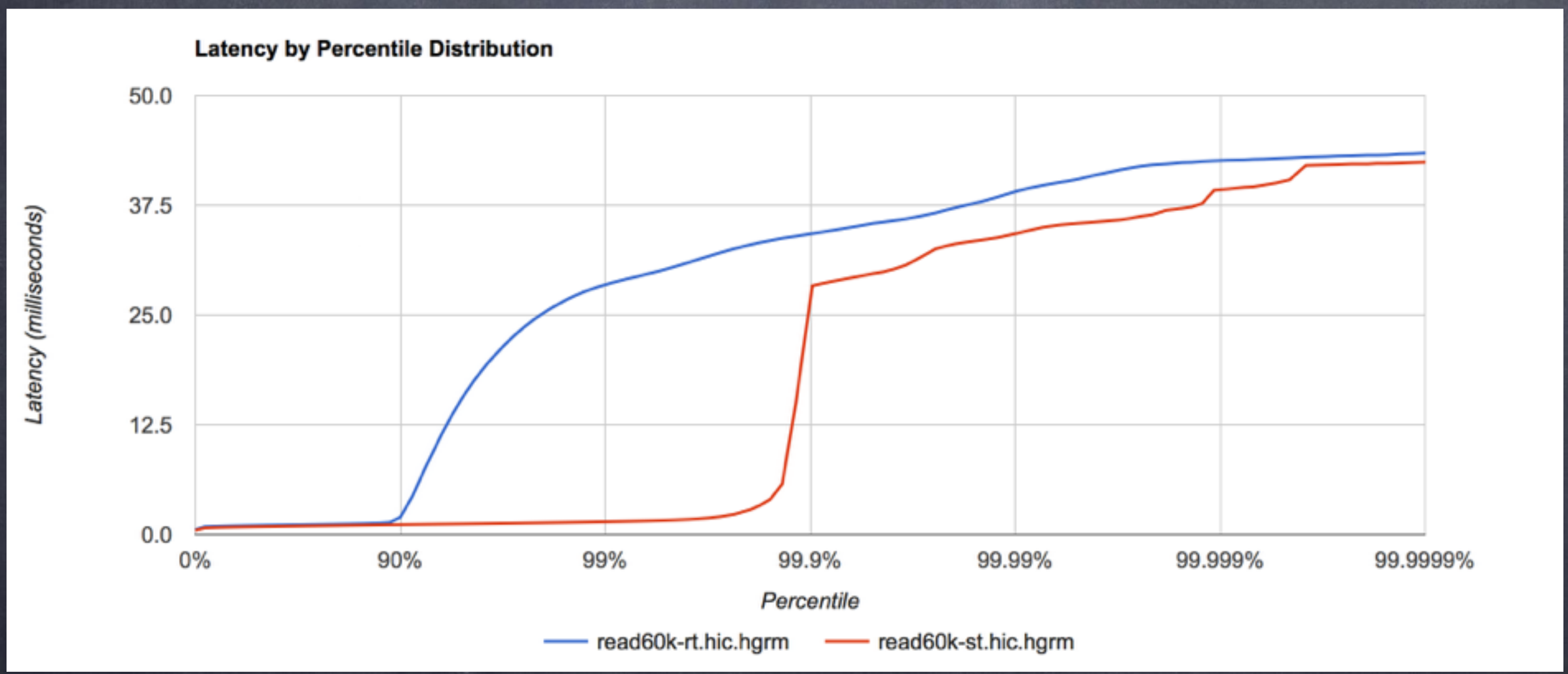
AZUL SYSTEMS

Response Time vs. Service Time @20K/sec



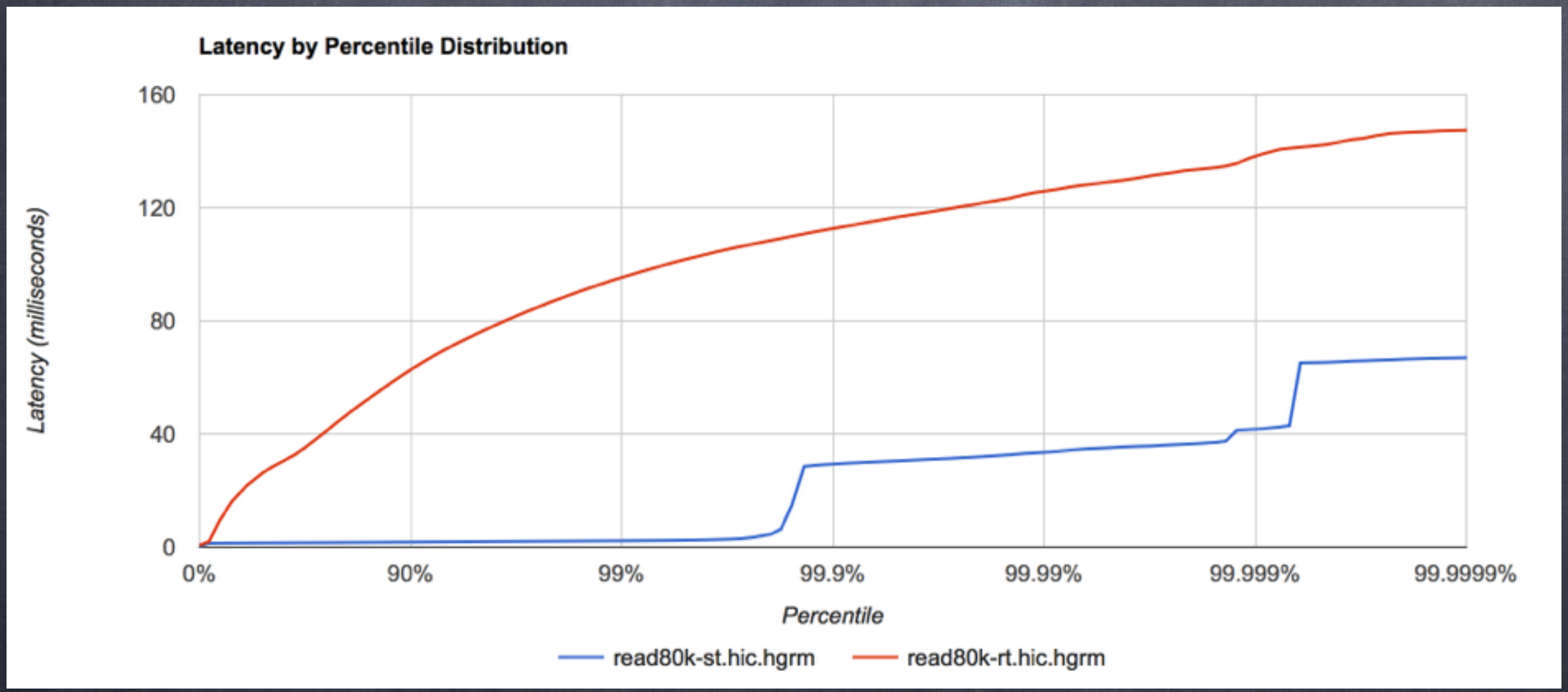
AZUL SYSTEMS

Response Time vs. Service Time @60K/sec



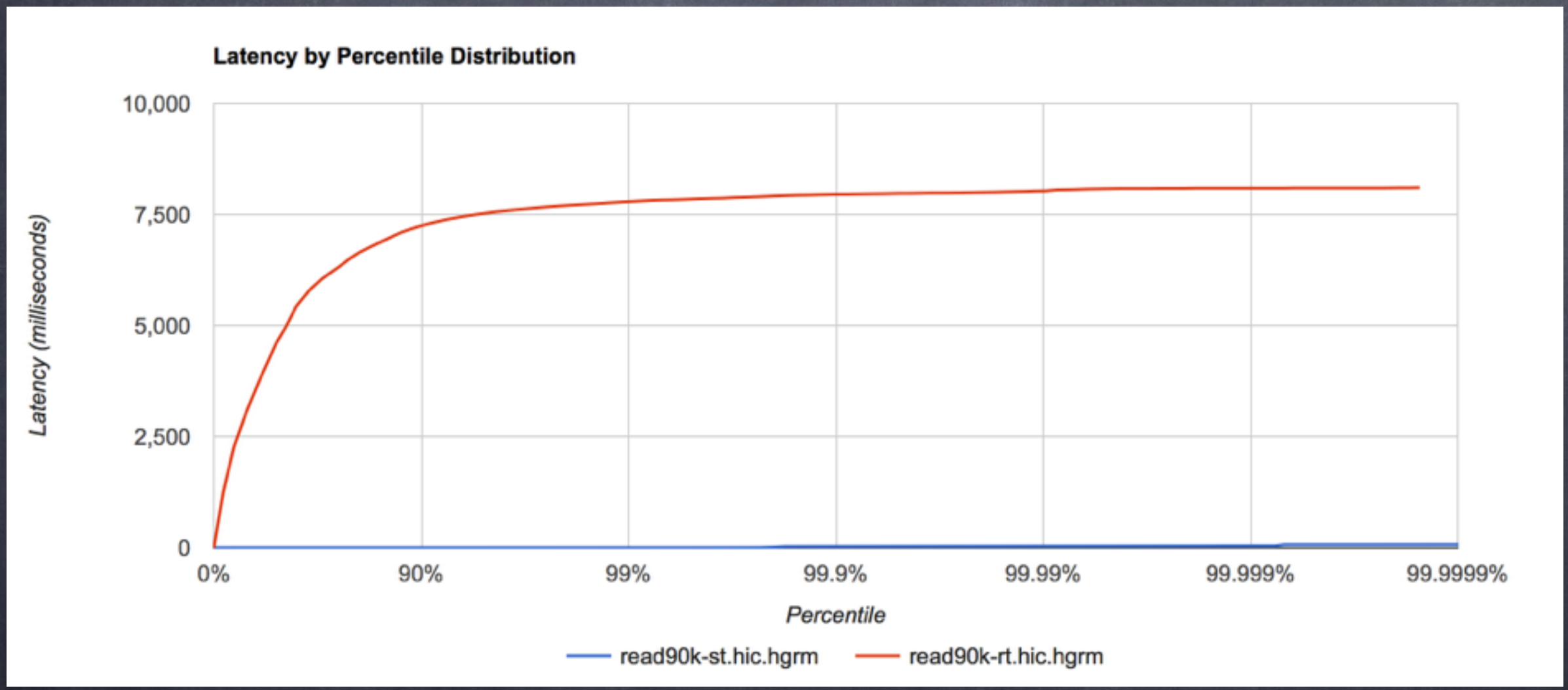
AZUL SYSTEMS

Response Time vs. Service Time @80K/sec



AZUL SYSTEMS

Response Time vs. Service Time @90K/sec

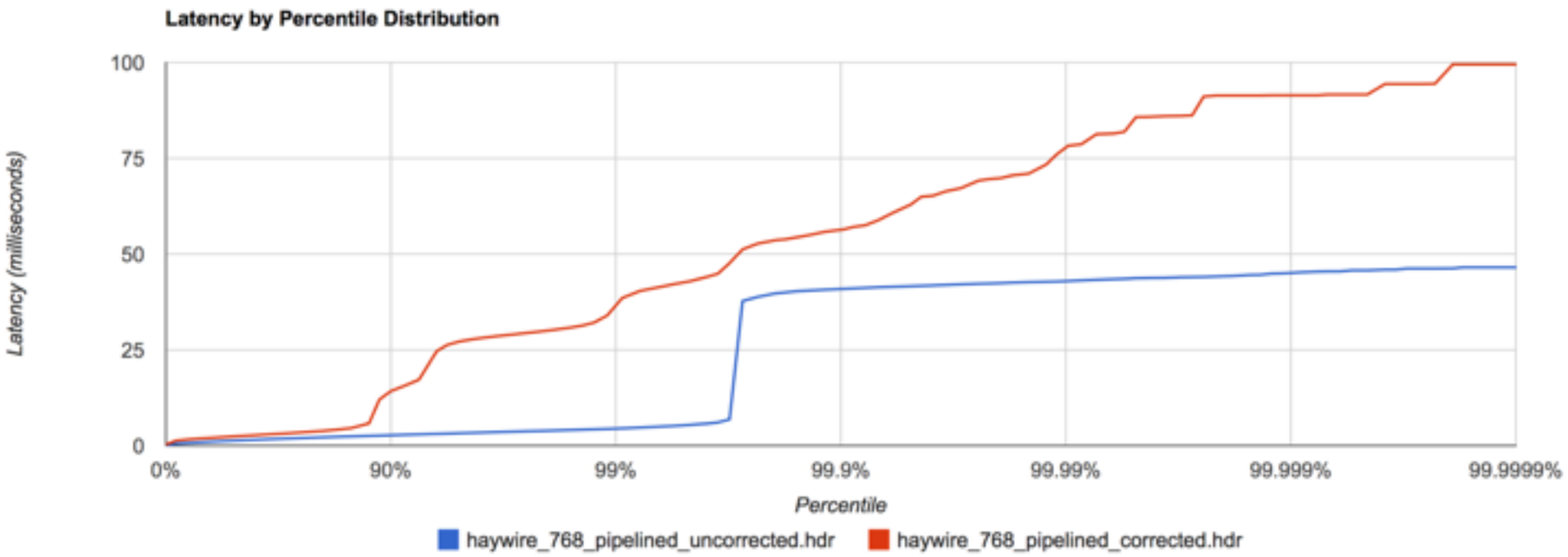


AZUL SYSTEMS

How "real" people react

Kelly Sommers @kellabyte 2d

LOL at how badly we all benchmark. Blue is how most of us are benchmarking, Red is the actual truth i.imgur.com/HYoWEu6.png

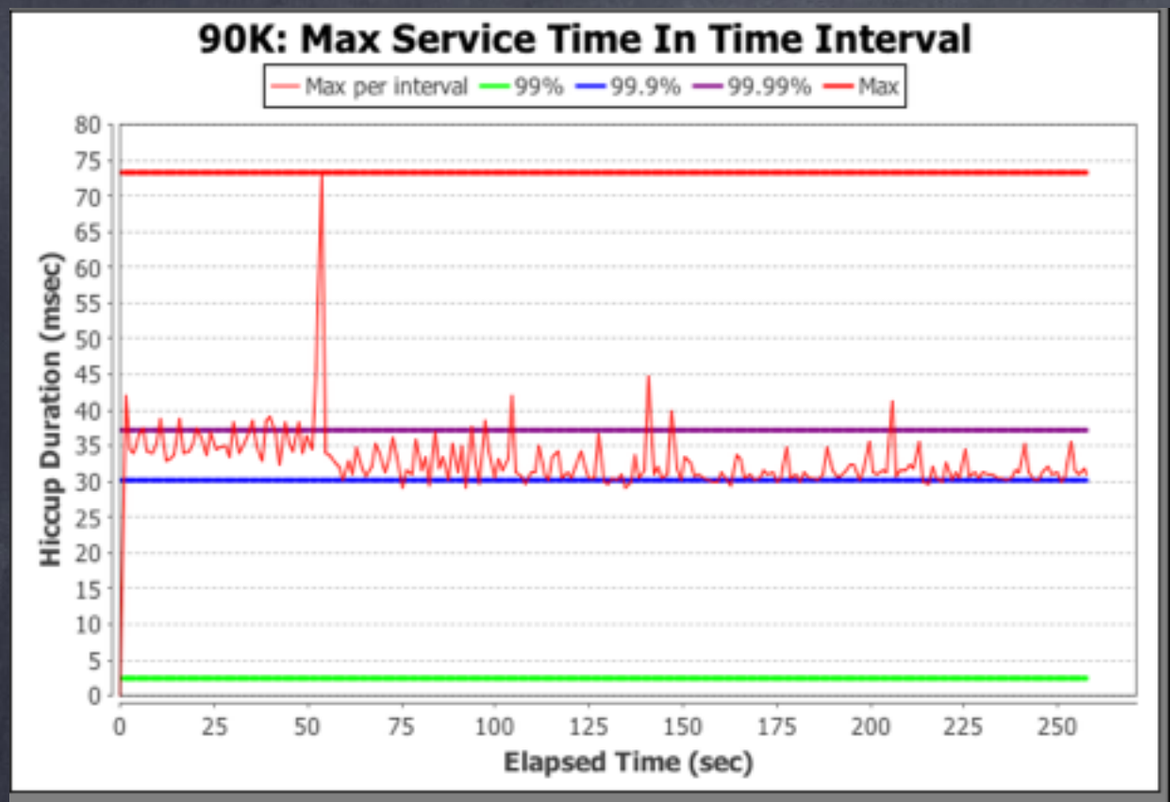


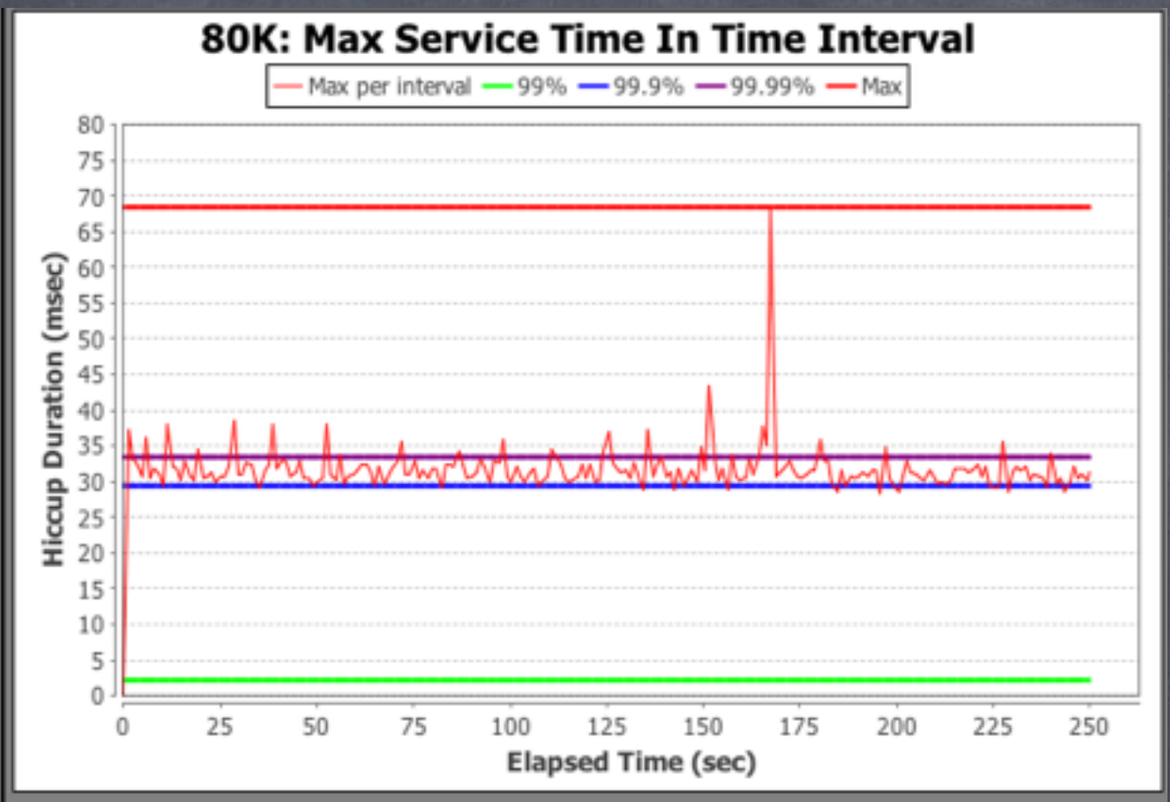
Leandro Pereira @lafp

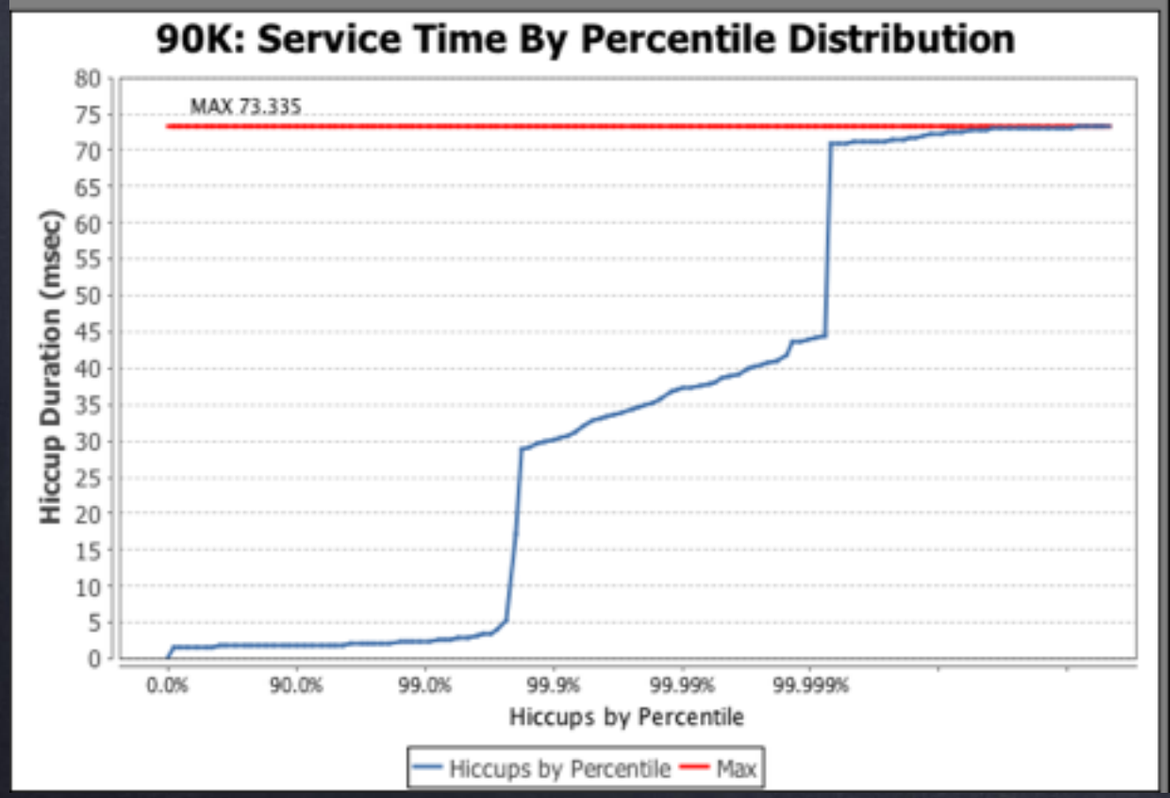
@kellabyte Blue, you believe in whatever you want to believe. Red, you wake up in Wonderland and see how deep the rabbit hole goes.

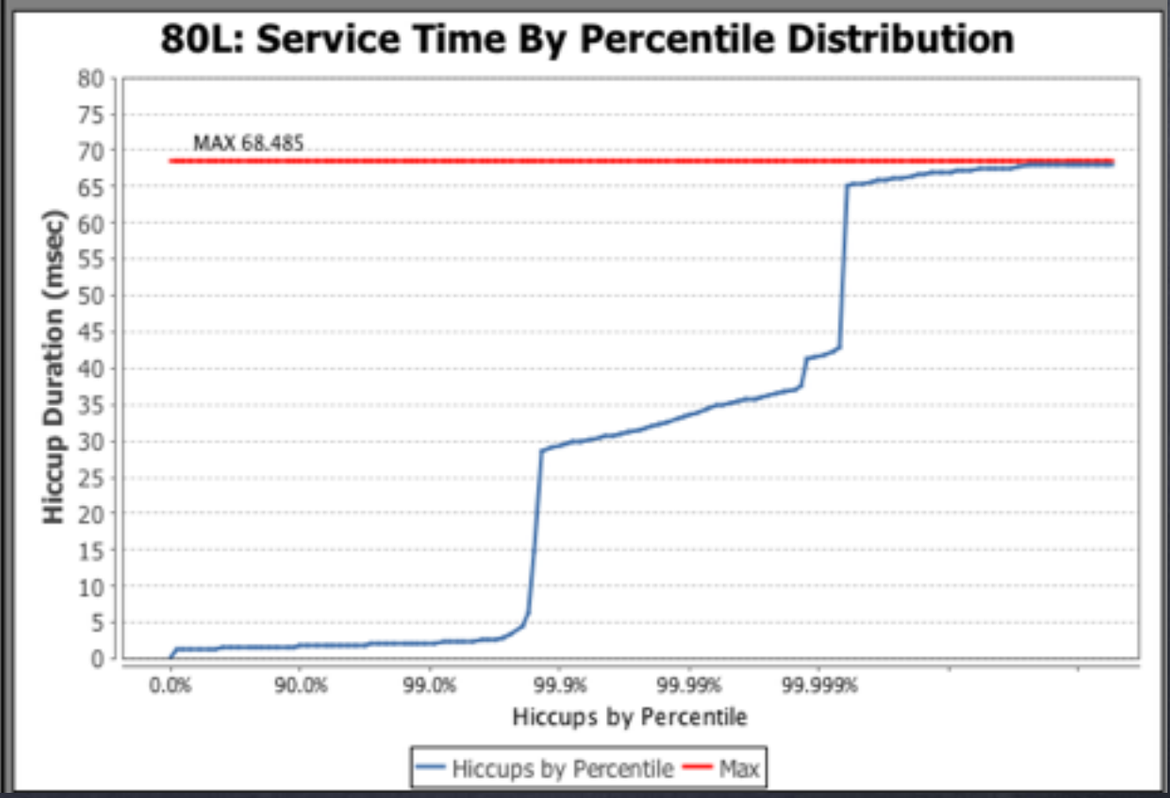
AZUL SYSTEMS

Service Time, 90K/s vs 80K/s



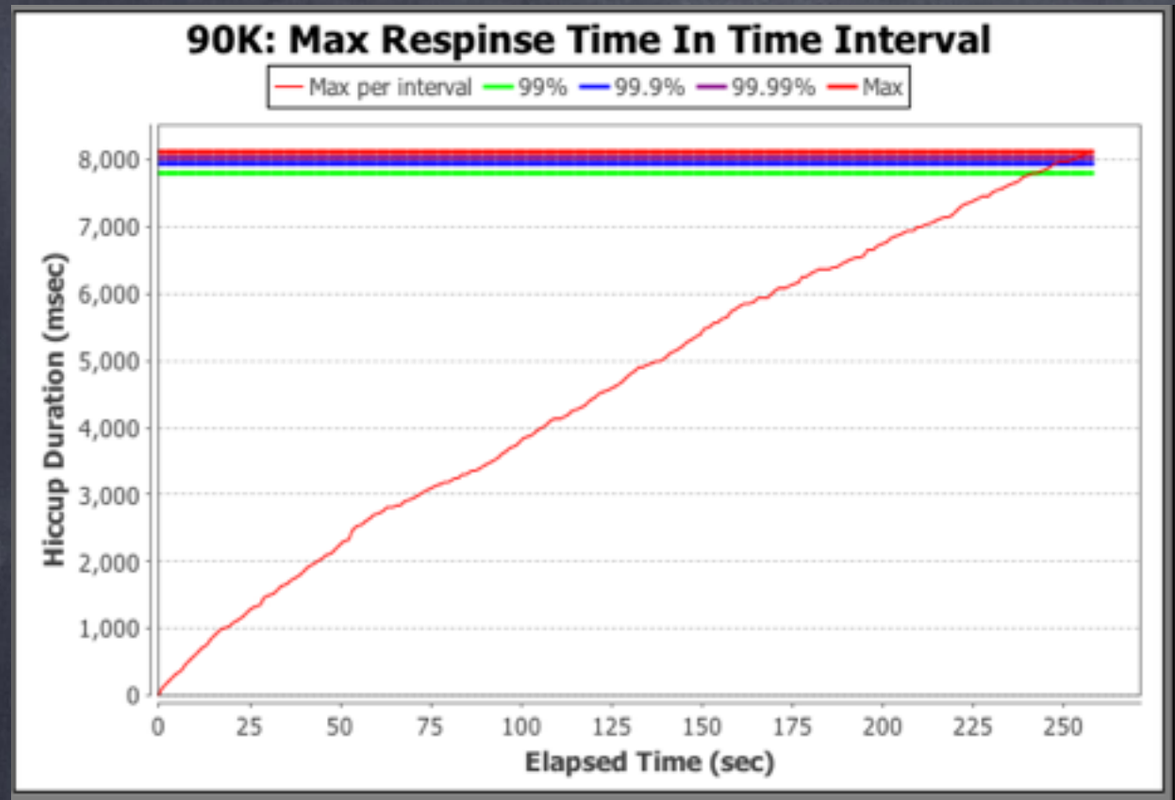


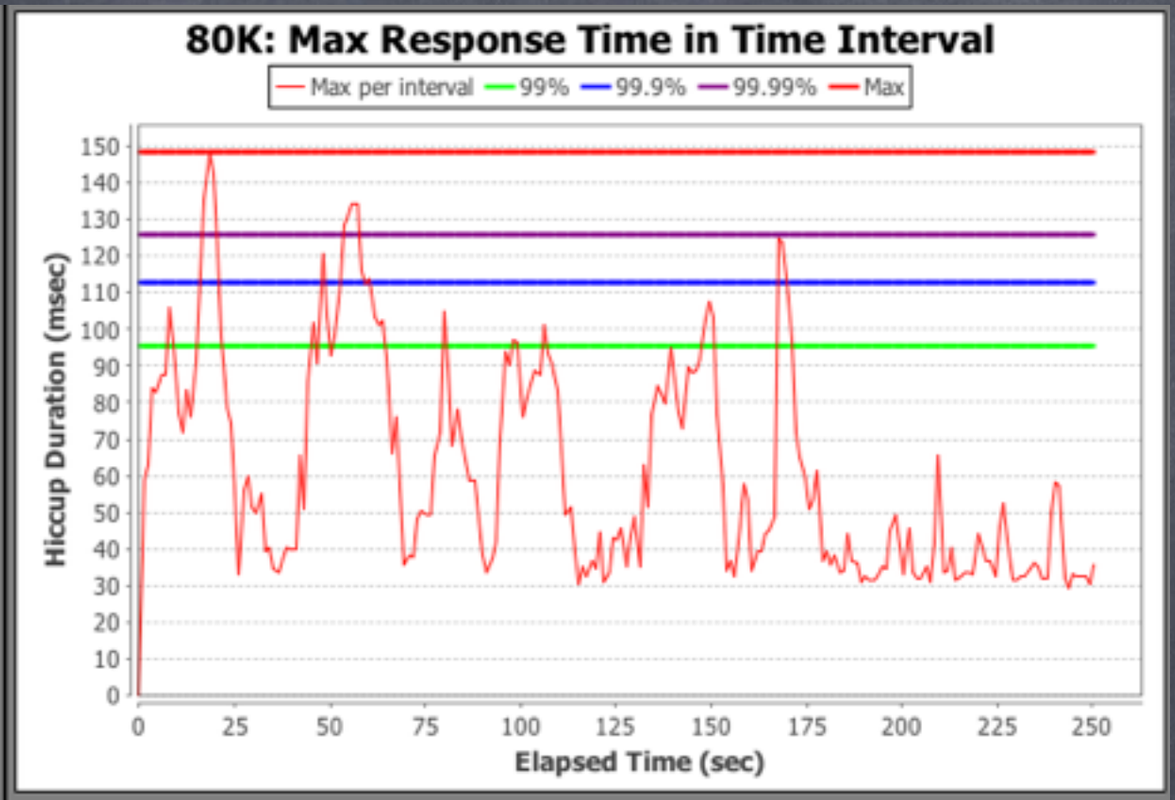


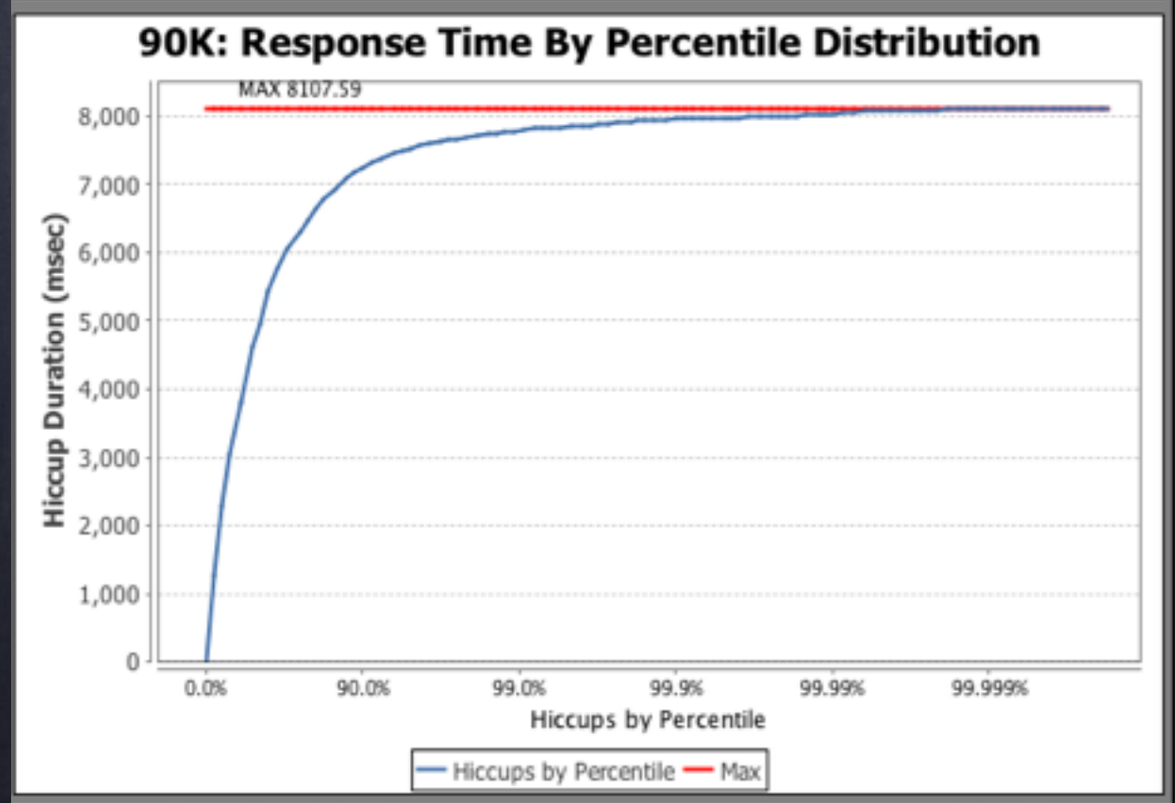


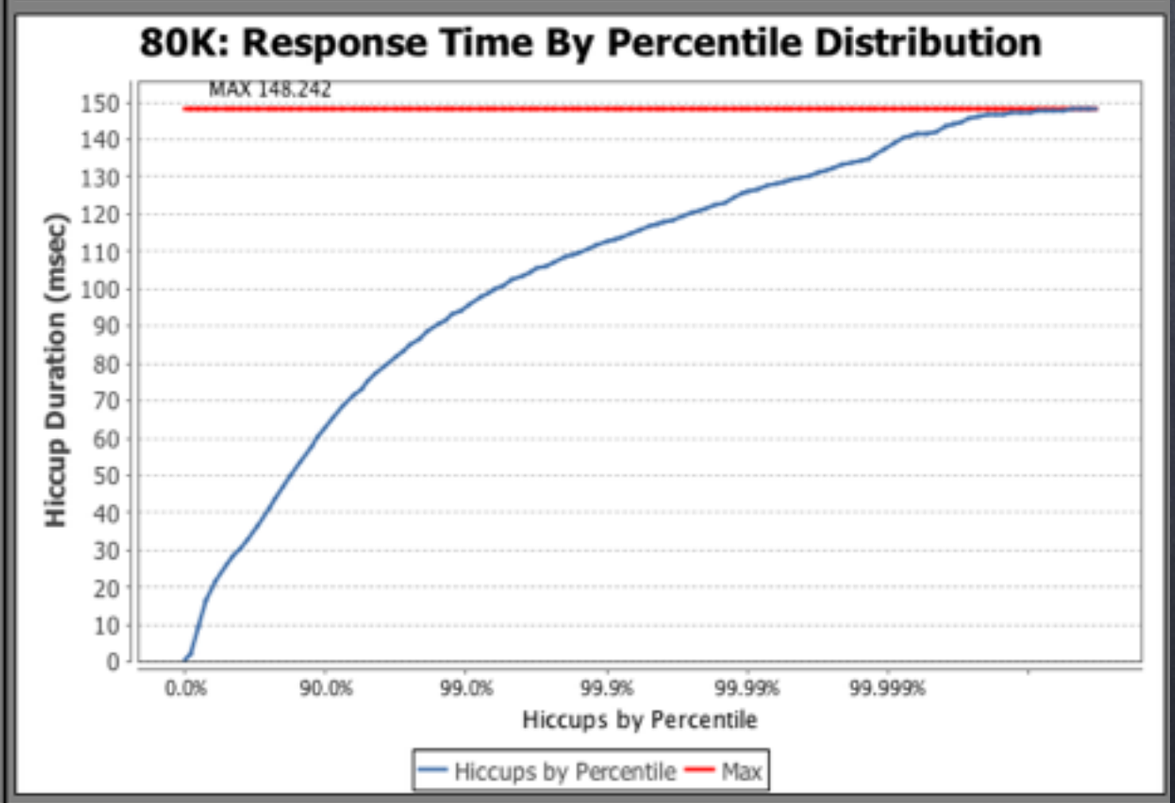
AZUL SYSTEMS

Response Time, 90K/s vs 80K/s



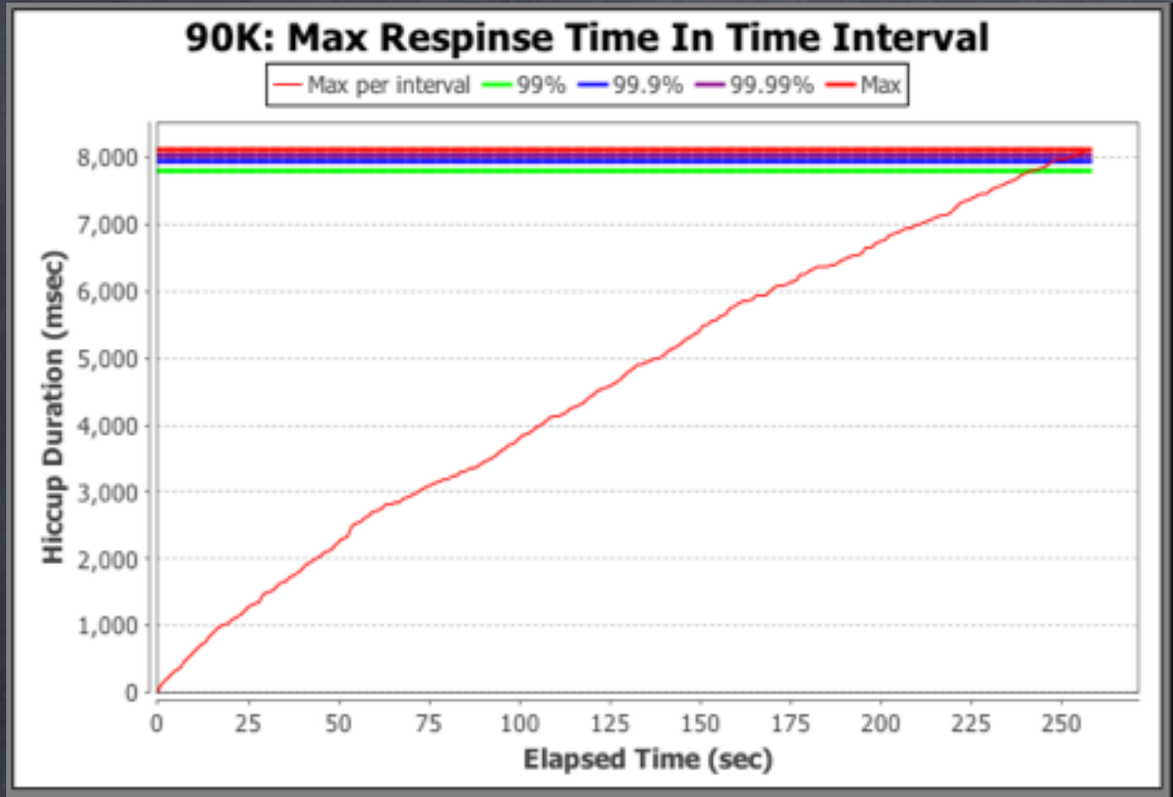


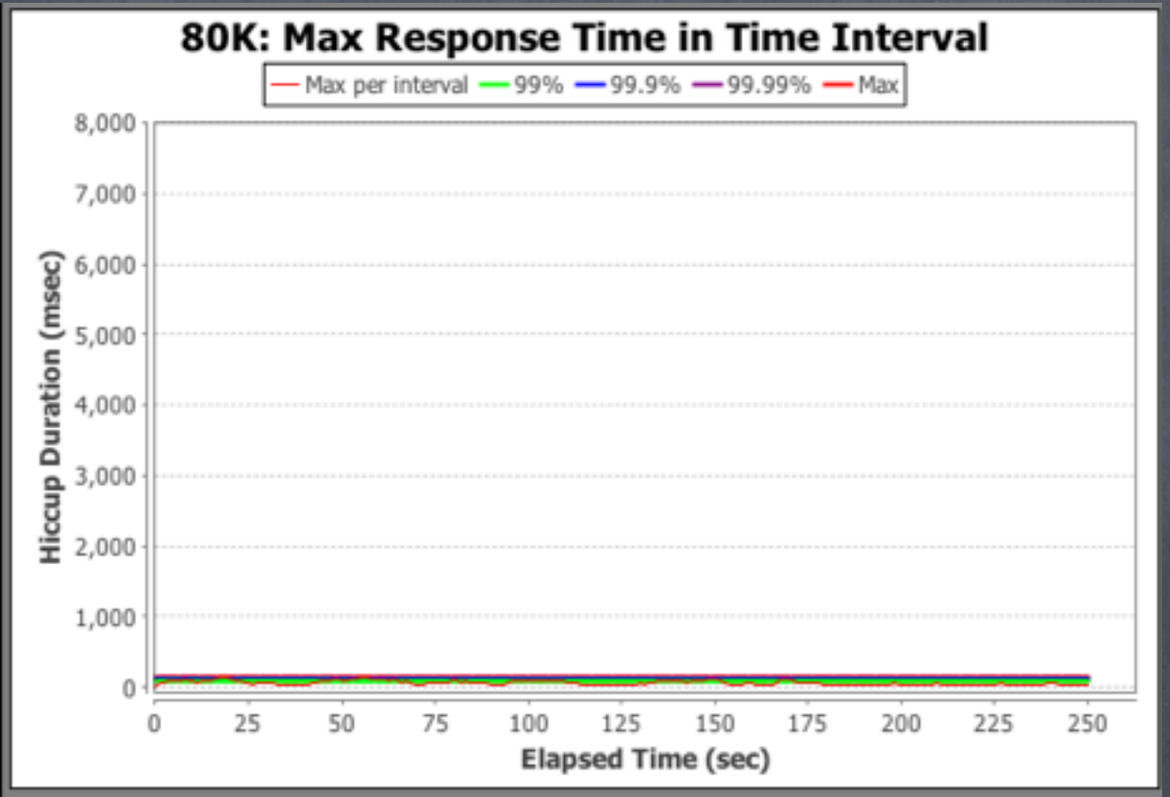


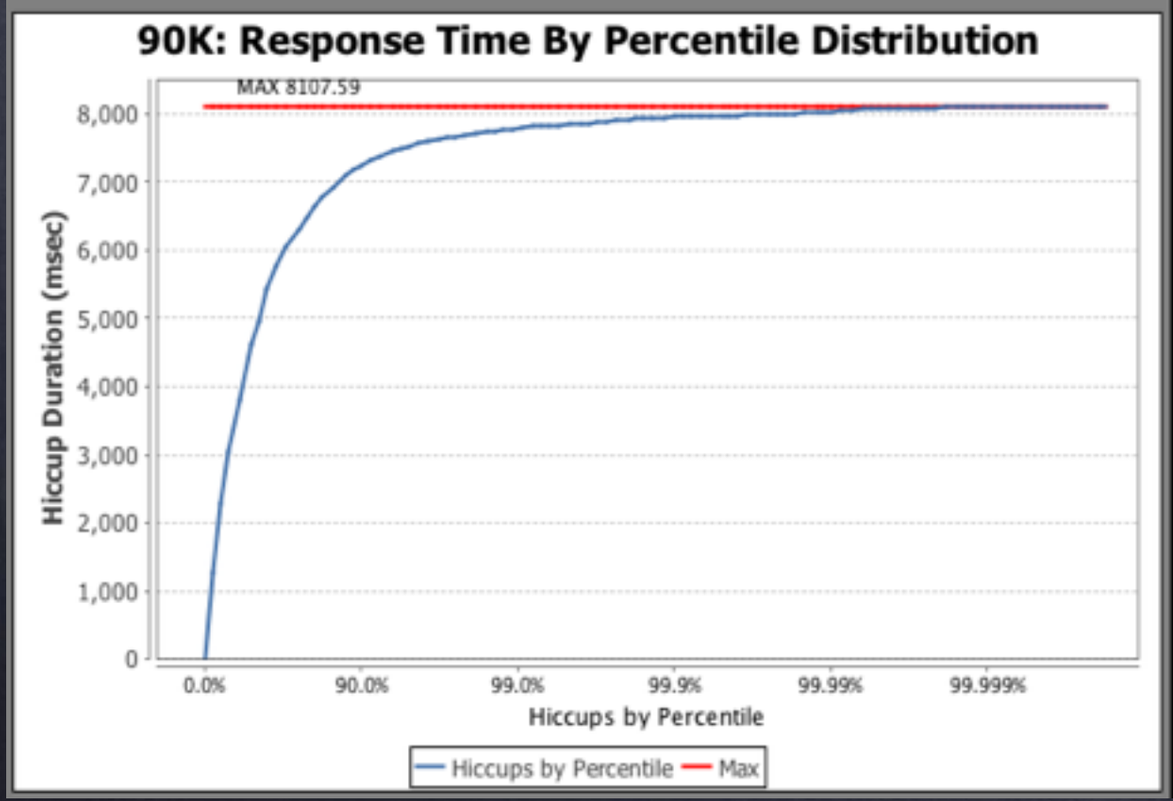


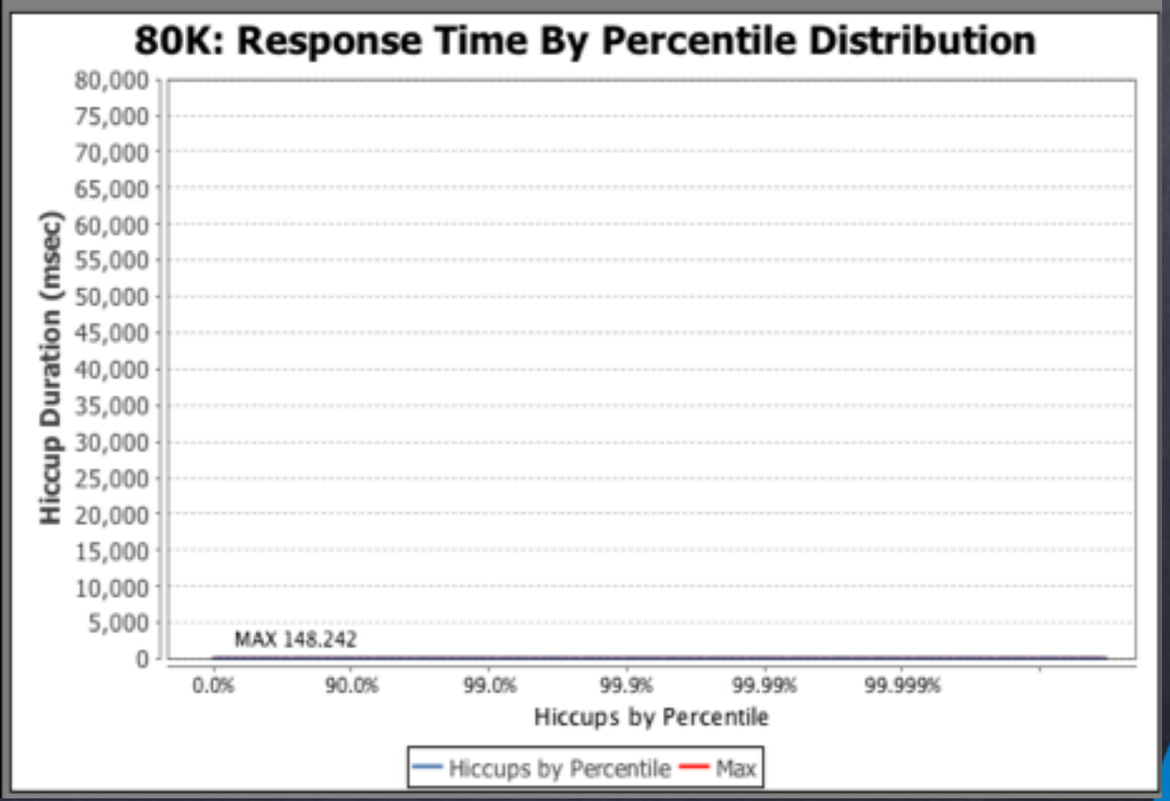
AZUL SYSTEMS

Response Time, 90K/s vs 80K/s (to scale)









AZUL SYSTEMS

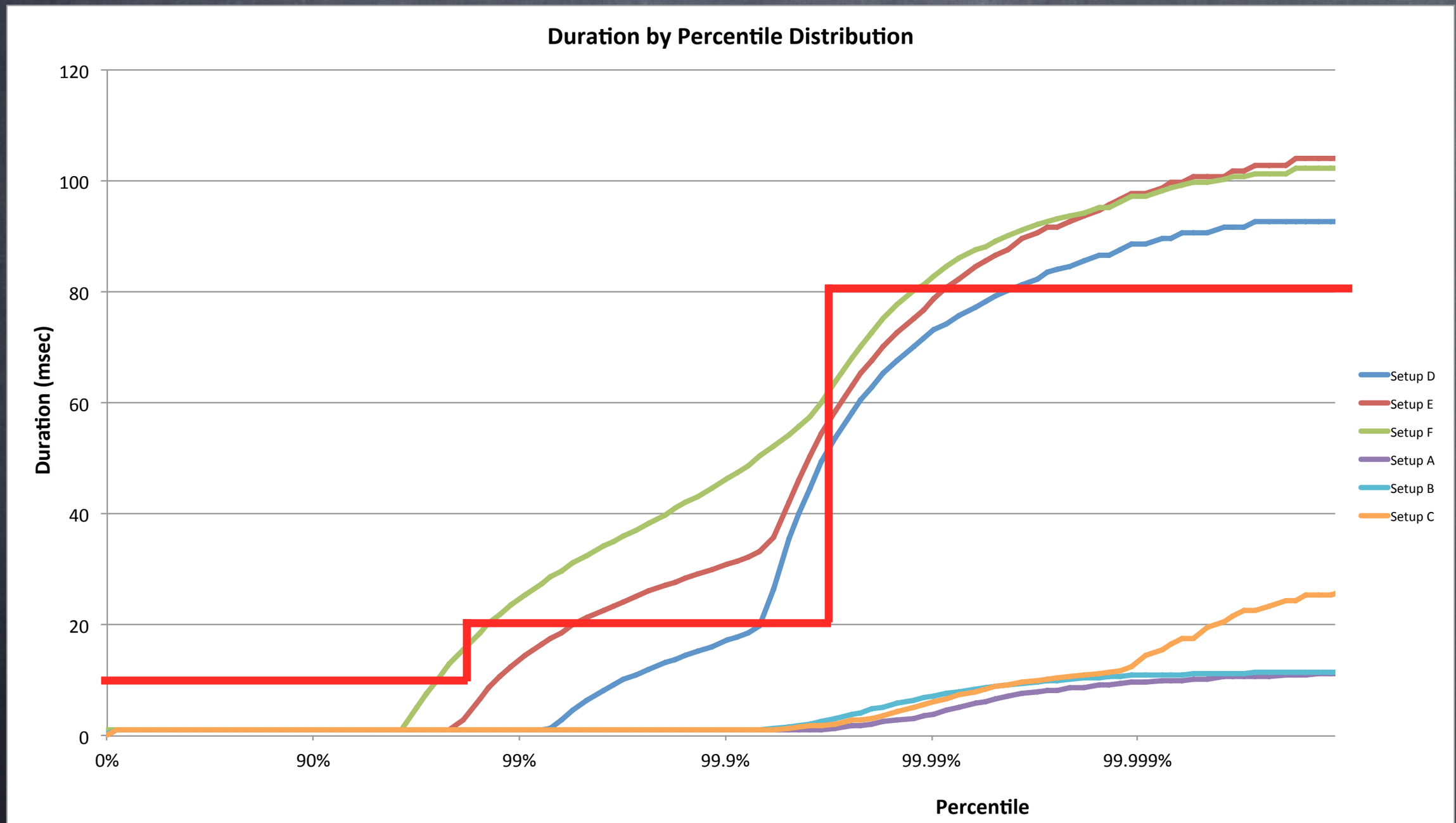
Latency doesn’t live in a vacuum

AZUL SYSTEMS

Sustainable Throughput:
The throughput achieved while safely maintaining service levels

AZUL SYSTEMS

Comparing behavior under different throughputs and/or configurations

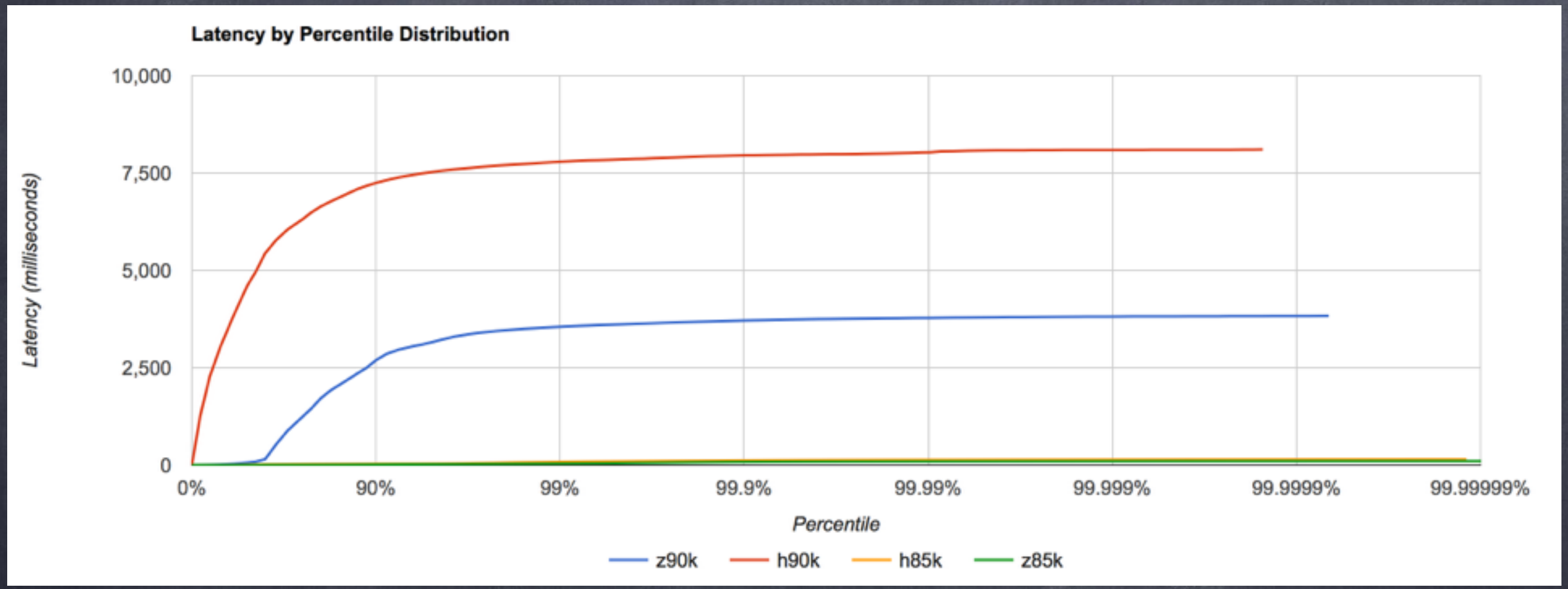


AZUL SYSTEMS

Comparing response time or latency behaviors

AZUL SYSTEMS

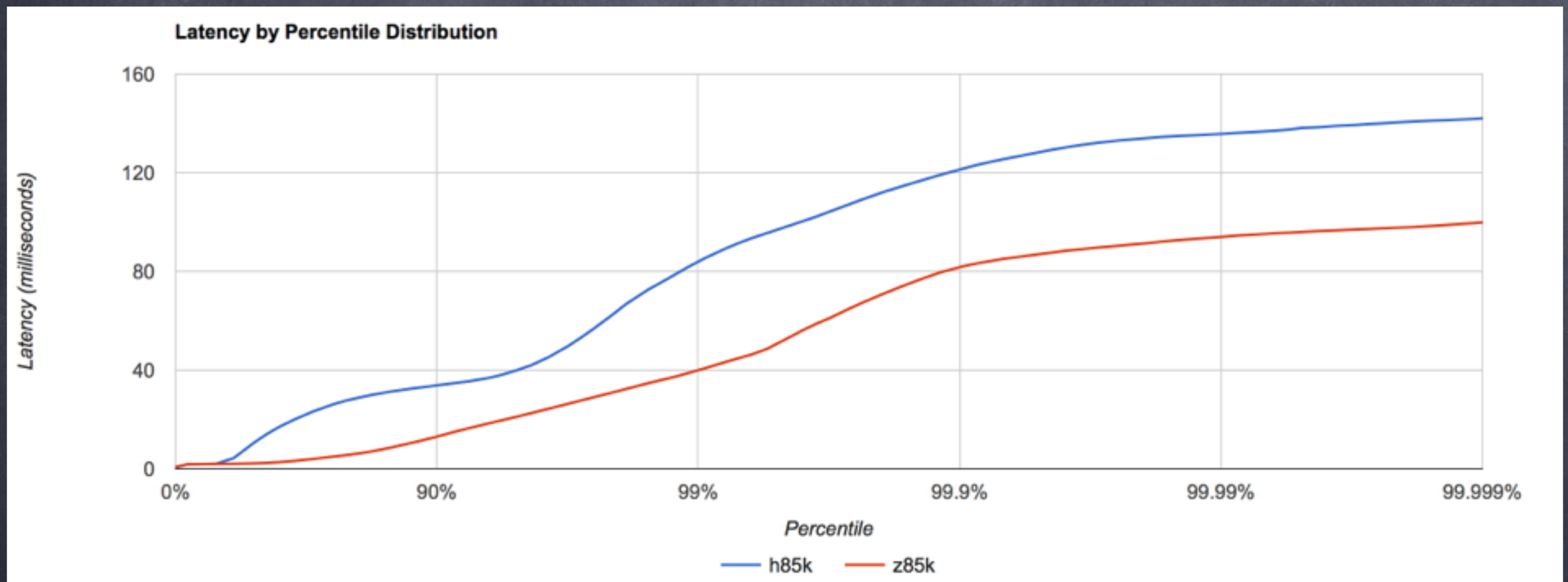
System A @90K/s & 85K/s vs. System B @90K/s & 85K/s



Wrong Place to Look: They both "suck" at >85K/sec

AZUL SYSTEMS

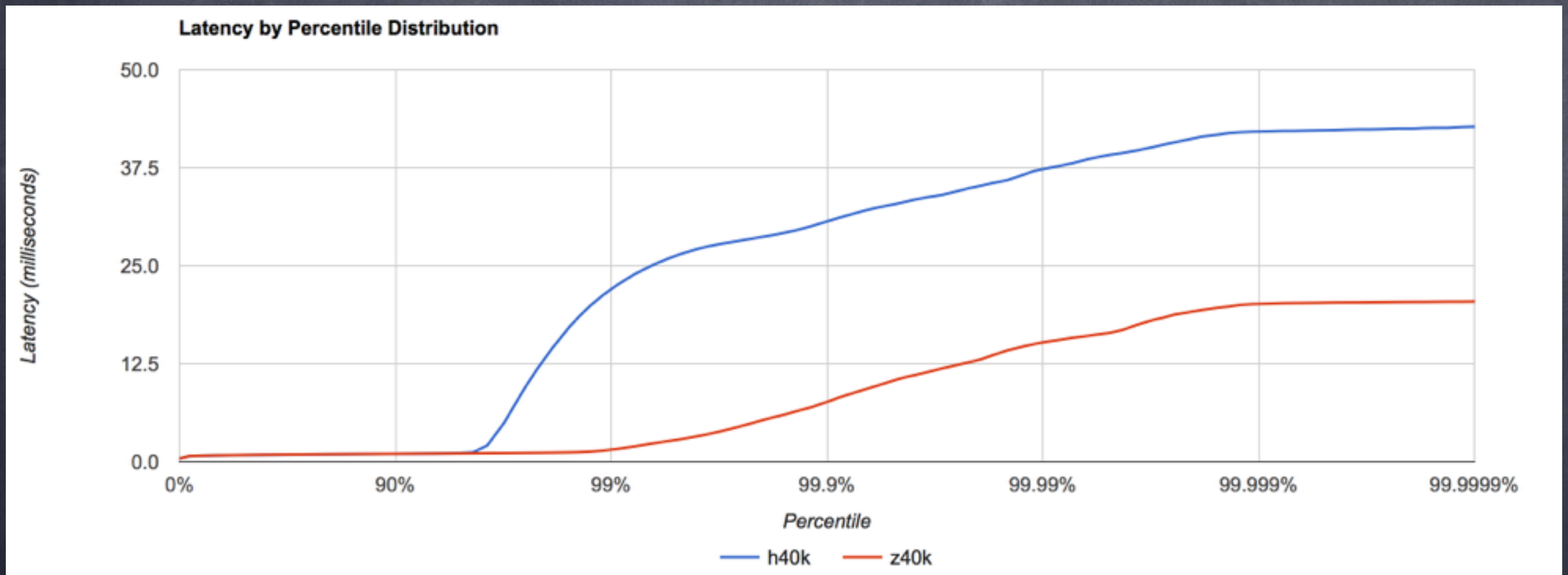
System A 85K/s vs. System B 85K/s



Looks good, but still
the wrong place to look

AZUL SYSTEMS

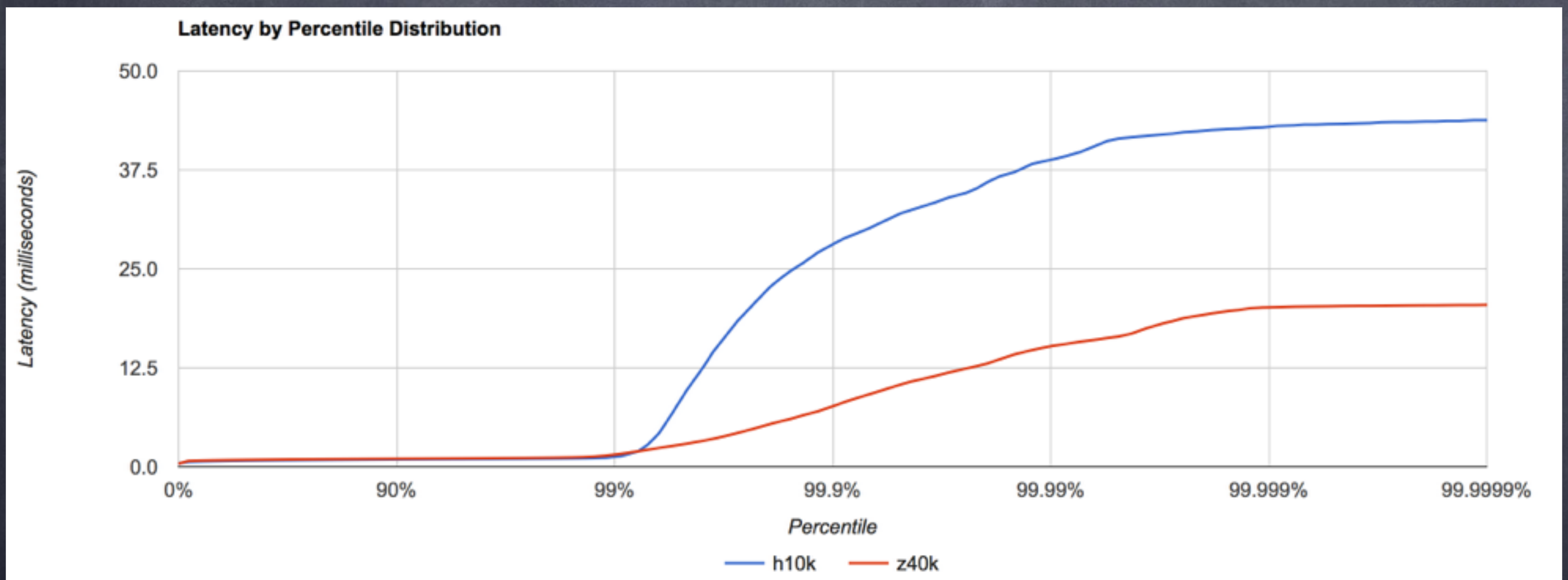
System A @40K/s vs. System B @40K/s



More interesting…
What can we do with this?

AZUL SYSTEMS

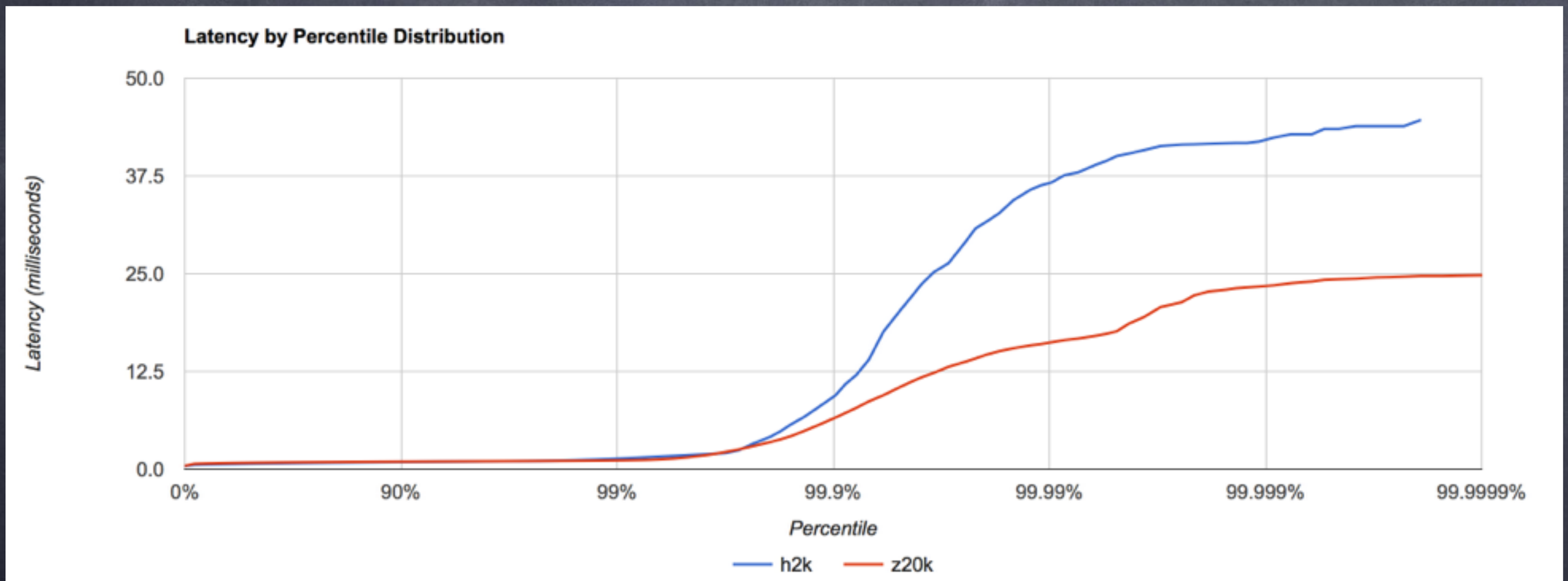
System A @10K/s vs. System B @40K/s



E.g. if “99%’ile < 5msec” was a goal:
System B delivers similar 99%’ile and superior 99.9%’ile+ while carrying 4x the throughput

AZUL SYSTEMS

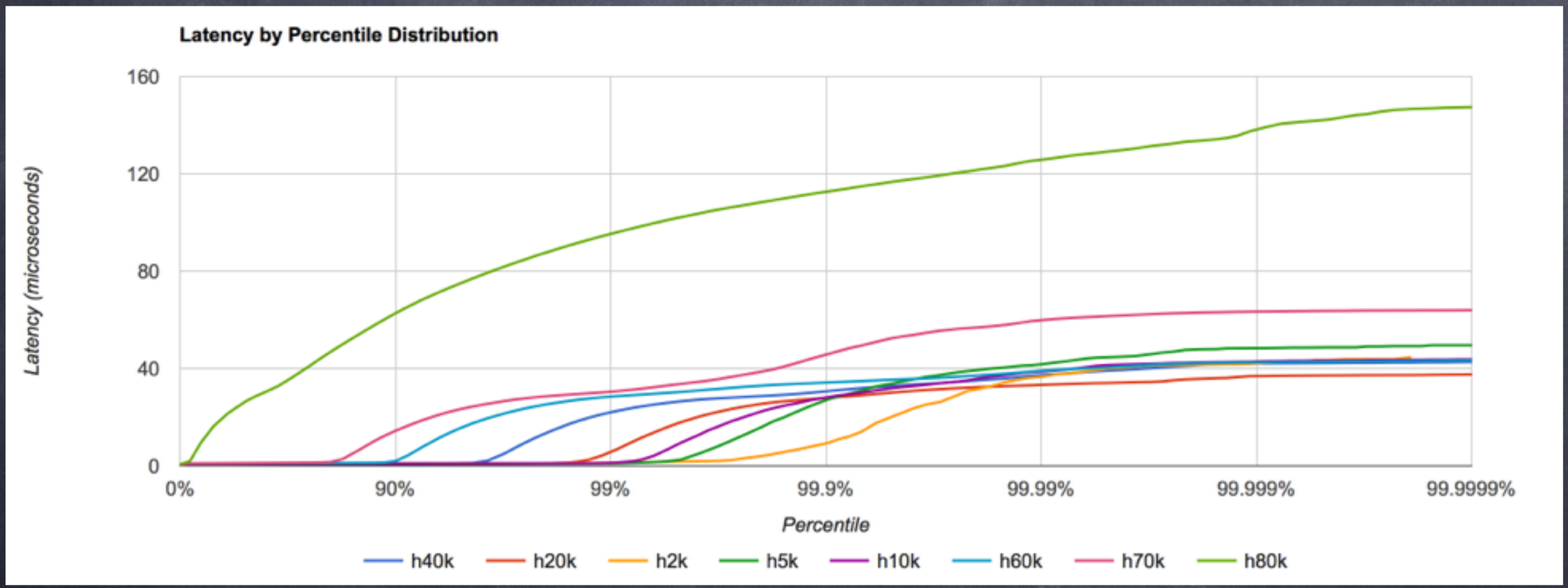
System A @2K/s vs. System B @20K/s



E.g. if “99.9%’ile < 10msec” was a goal:
System B delivers similar 99%’ile and 99.9%’ile
while carrying 10x the throughput

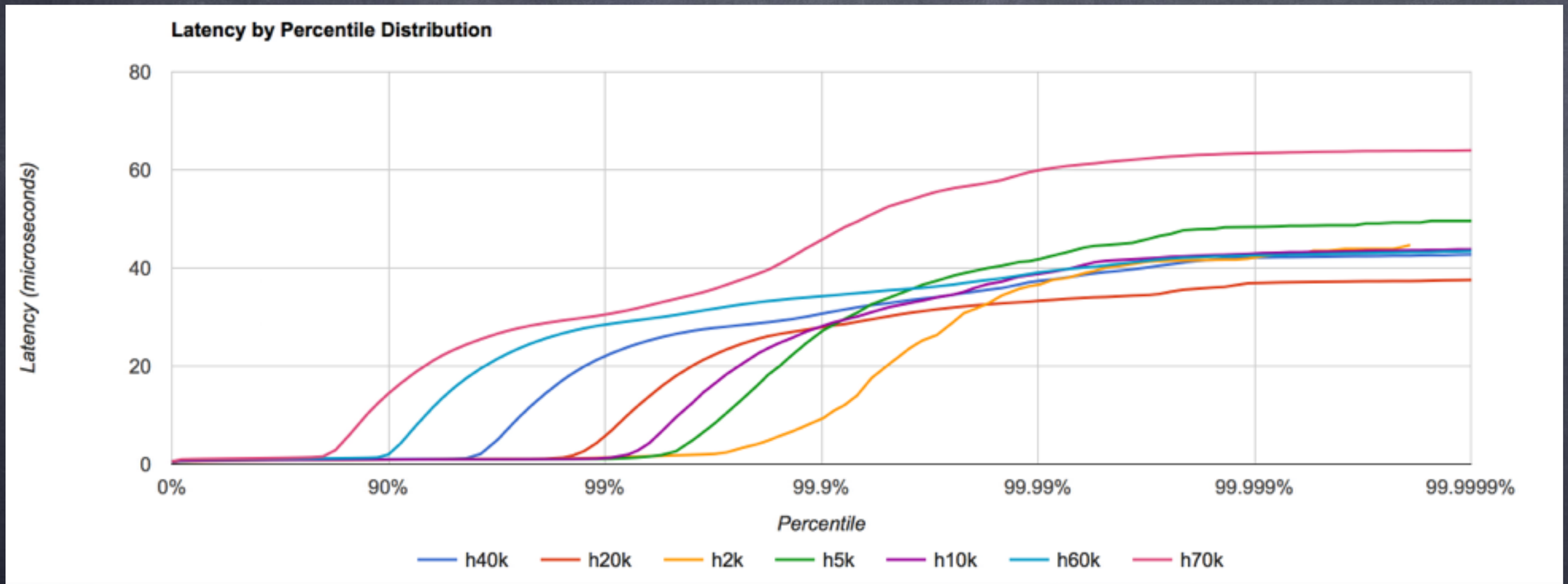
AZUL SYSTEMS

System A @2k thru 80k



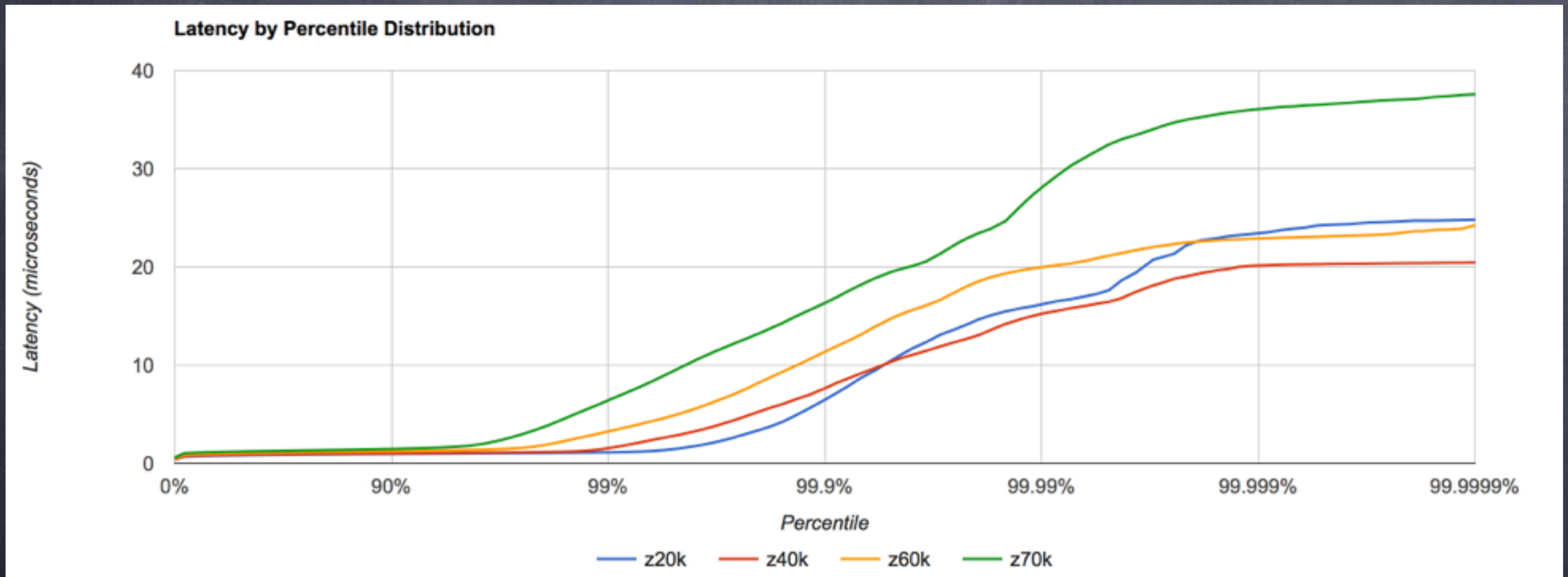
AZUL SYSTEMS

System A @2k thru 70k



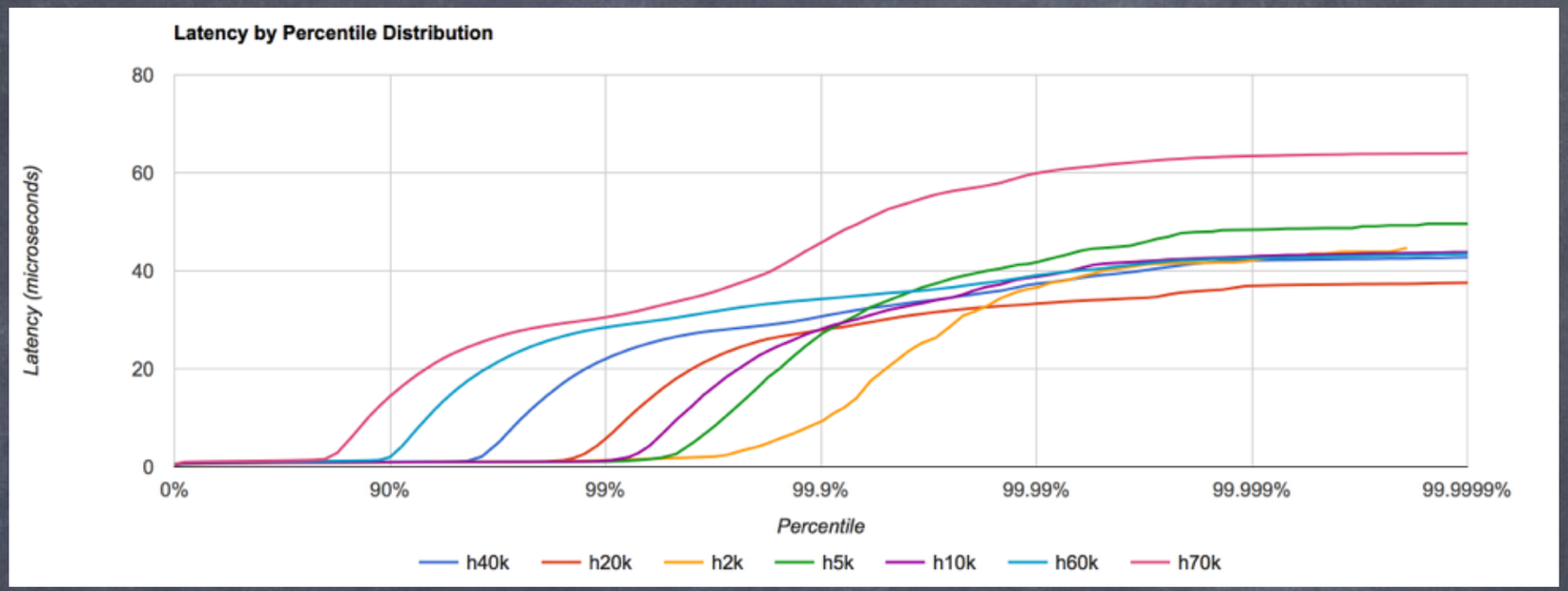
AZUL SYSTEMS

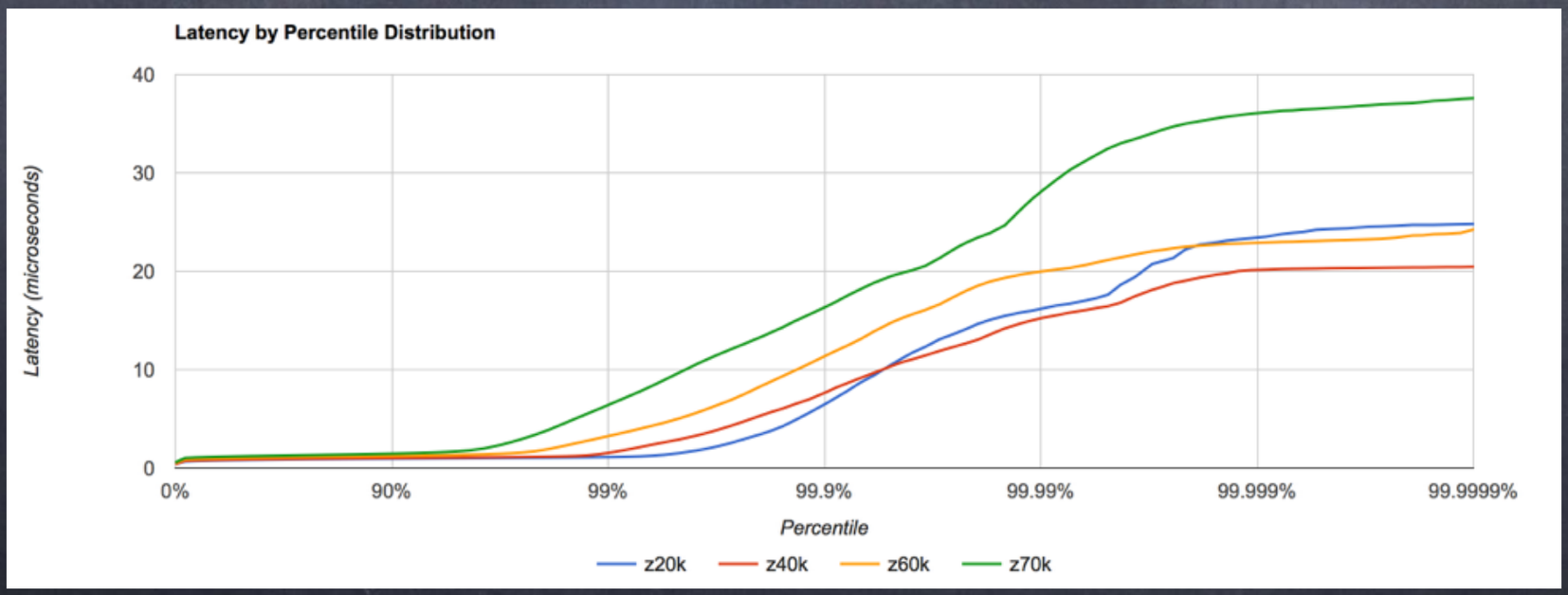
System B @20k thru 70k



AZUL SYSTEMS

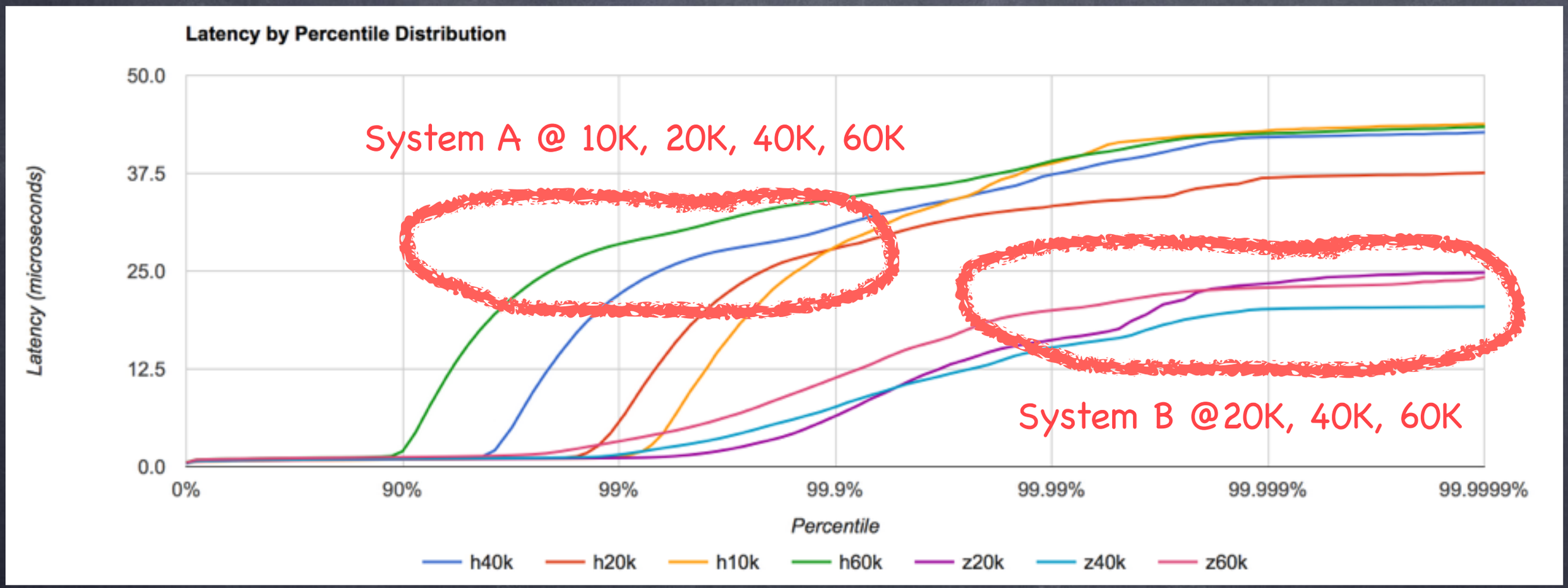
System A & System B @2k thru 70k





AZUL SYSTEMS

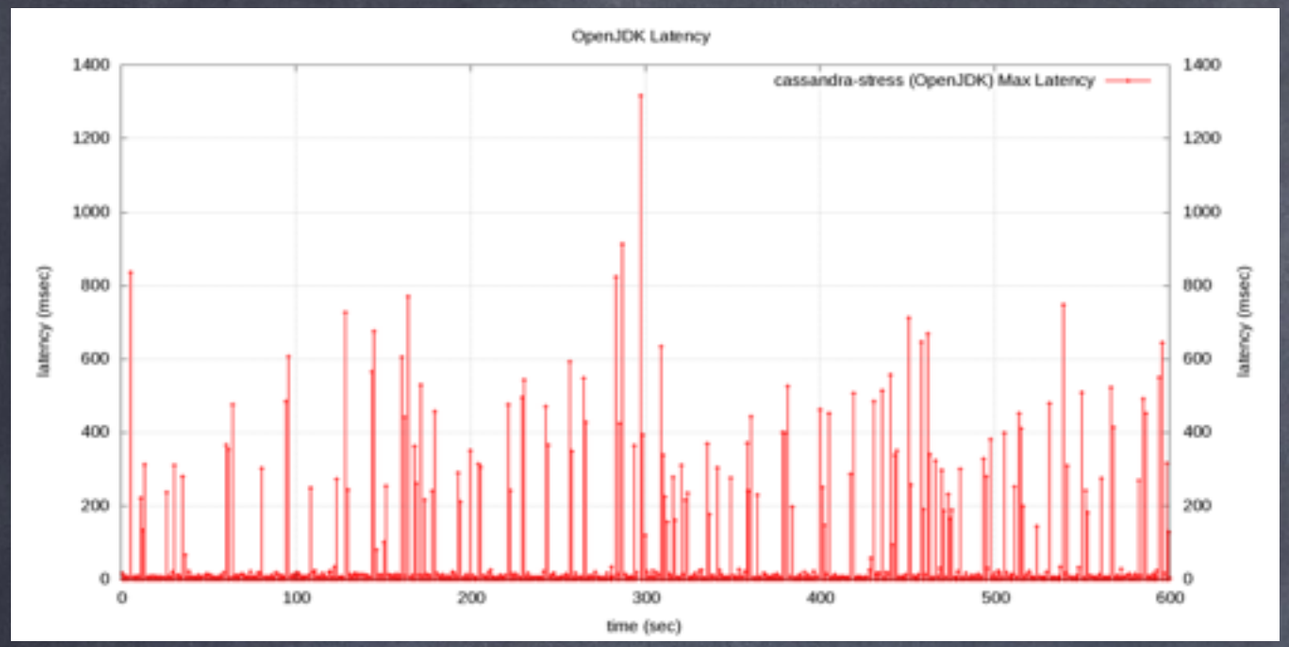
System A & System B
10K/s thru 60K/s



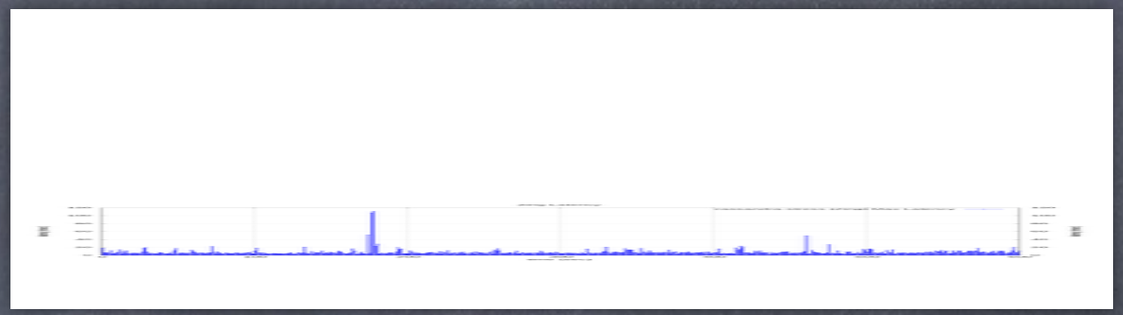
Lots of conclusions can be drawn from the above…
E.g. System B delivers a consistent 100x reduction in the rate of occurrence of >20msec response times

AZUL SYSTEMS

System A: 200-1400 msec stalls



System B drawn to scale



```
op rate                   : 40001
partition rate            : 26996
row rate                  : 26996
latency mean              : 30.6 (0.7)
latency median            : 0.5 (0.5)
latency 95th percentile   : 244.4 (1.1)
latency 99th percentile   : 537.4 (2.0)
latency 99.9th percentile : 1052.2 (8.4)
latency max               : 1314.9 (1312.8)
```

Response Time Service time

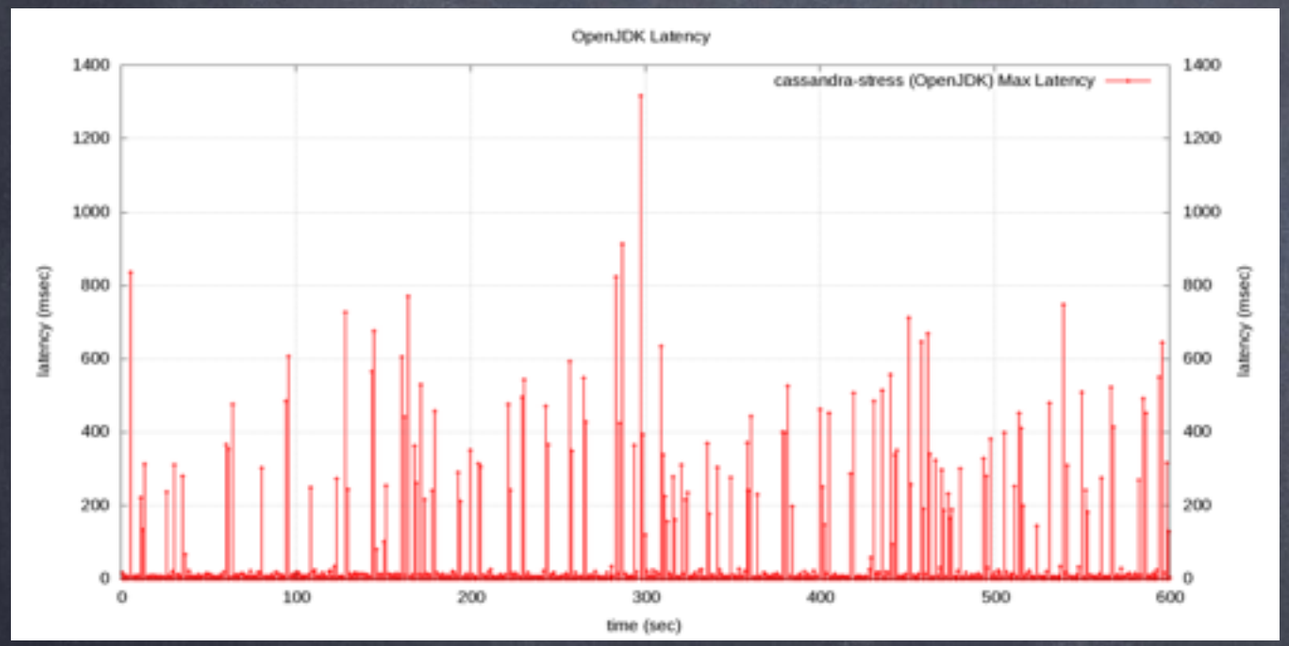
```
op rate                   : 40001
partition rate            : 26961
row rate                  : 26961
latency mean              : 0.6 (0.5)
latency median            : 0.5 (0.5)
latency 95th percentile   : 1.0 (0.9)
latency 99th percentile   : 2.7 (1.9)
latency 99.9th percentile : 13.3 (3.8)
latency max               : 110.6 (28.2)
```

Response Time Service time

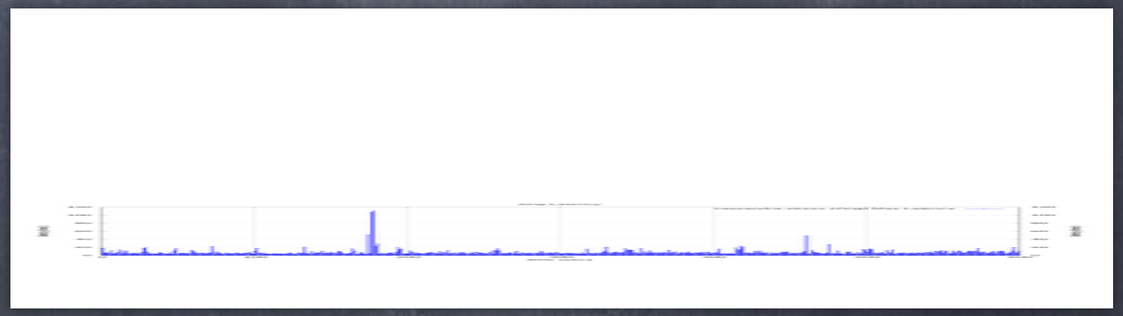
AZUL SYSTEMS

A simple visual summary

This is Your Load on System A



This is Your Load on System B



Any Questions?

AZUL SYSTEMS