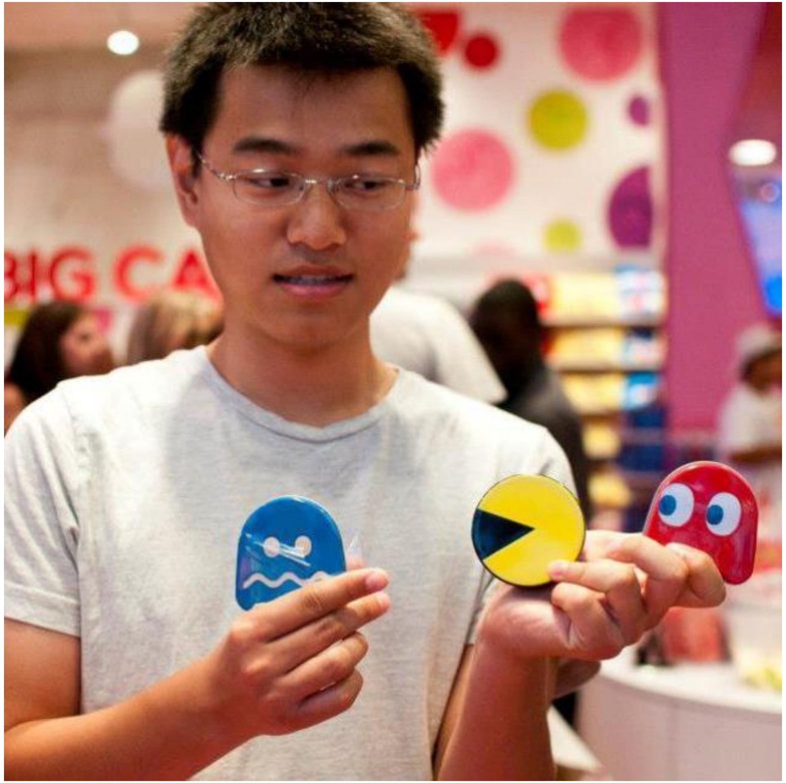
Did Kubernetes Make My P95s Worse?

Jian Cheung & Stephen Chan • KubeCon Nov 2019

Who are we?

Who are we?

Hi, I'm Jian!



Hi, I'm Stephen!

Outline

- Brief intro of Kubernetes at Airbnb
- Dive in to some cases
    - Latencies *Improved*?
    - Noisy Neighbors
    - Noisy Neighbors, made worse by Kubernetes
    - Write Once, Run Anywhere
    - Traffic Imbalance
    - Kube DNS slowness
- Recap

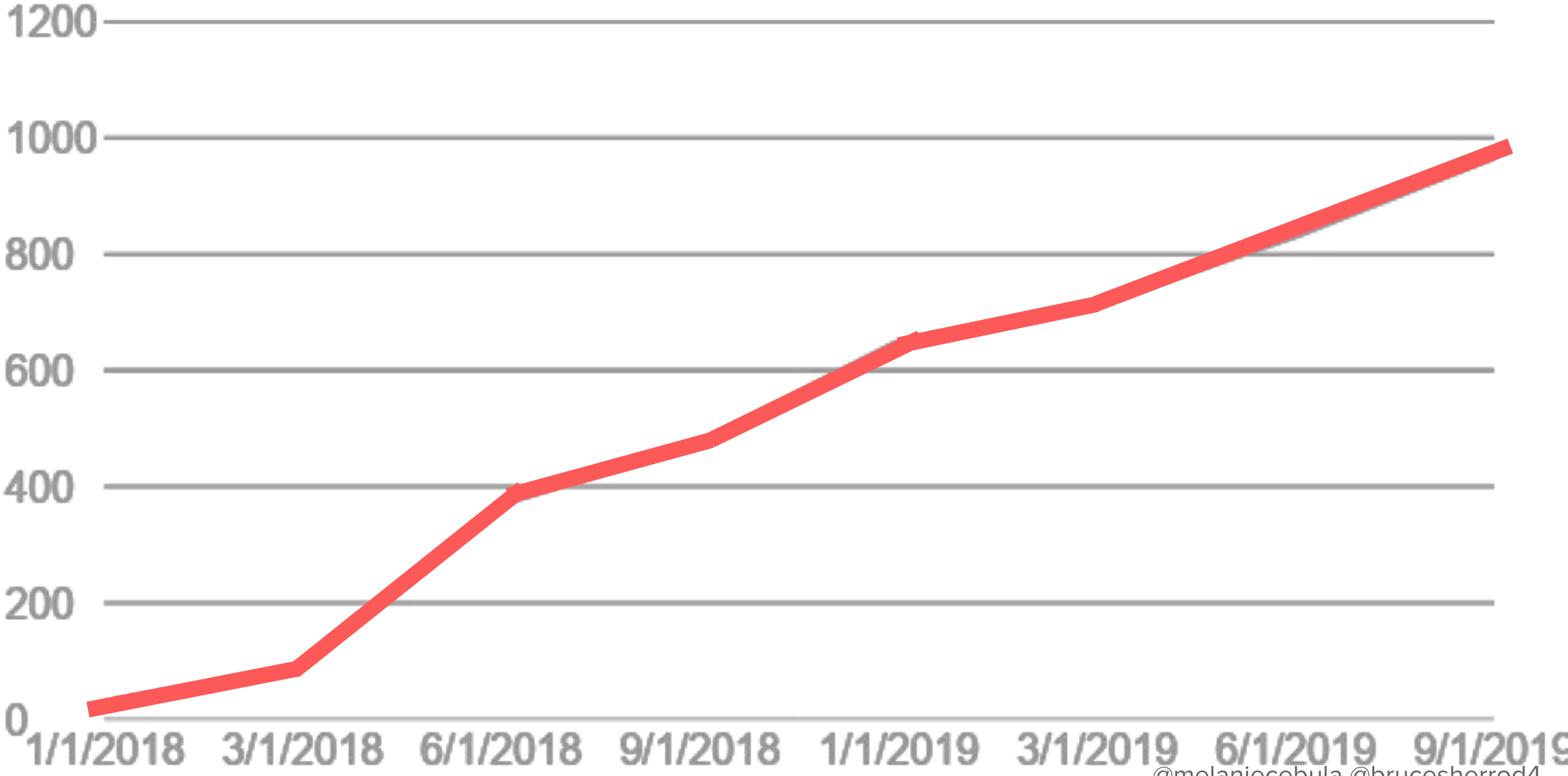
And Containers

Kubernetes at Airbnb

@jiancheung & Stephen Chan

Kubernetes @Airbnb

SERVICES



@melaniecebula @brucesherrod4

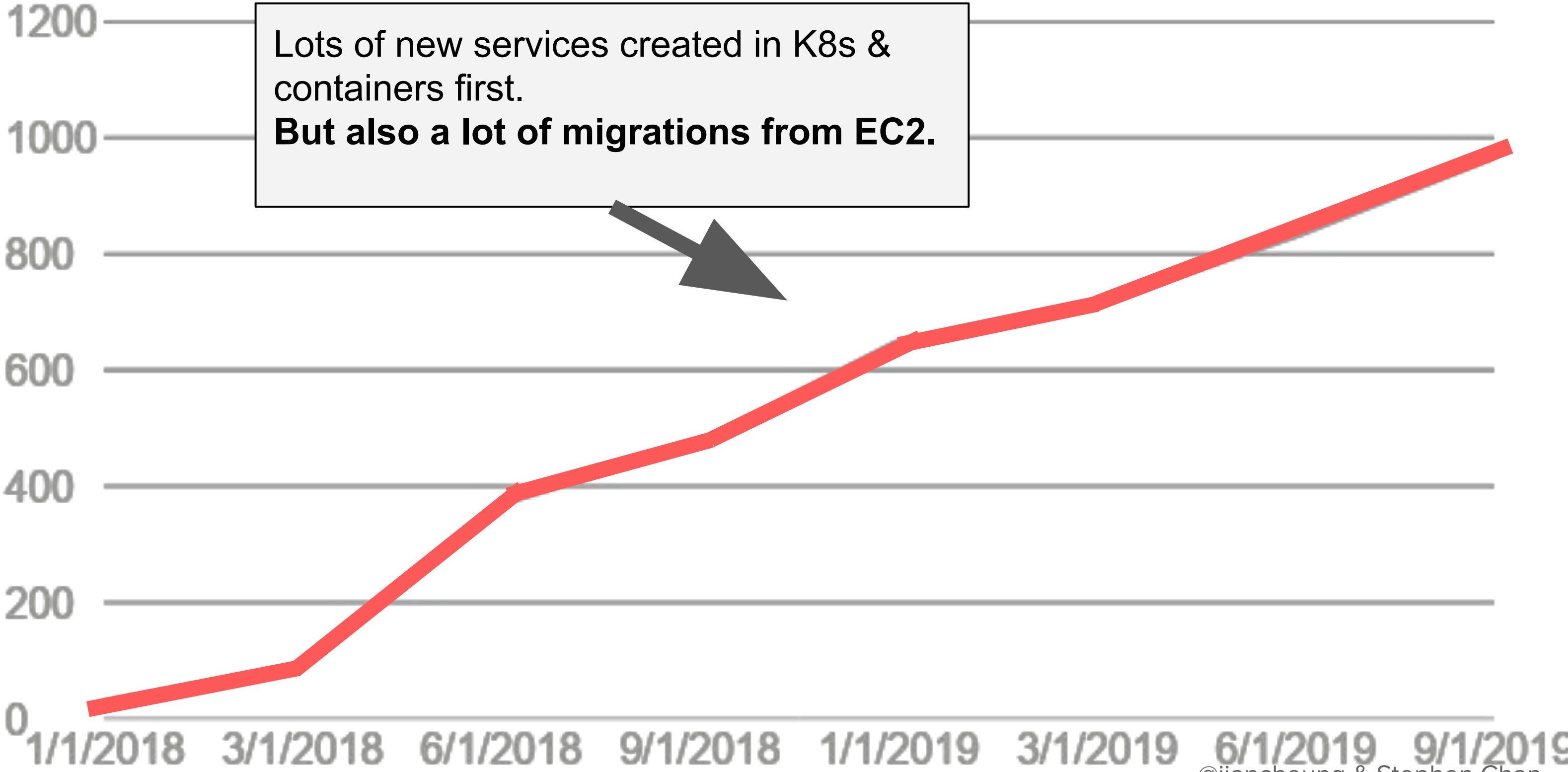
Airbnb Kubernetes Environment

- Amazon Linux 2
- Ubuntu Images
- Canal (Calico + Flannel) CNI plugin
- Nodeport services/Smartstack
- Many languages (ruby, java, python, go, etc)

.. and then the problems

Kubernetes @Airbnb

SERVICES

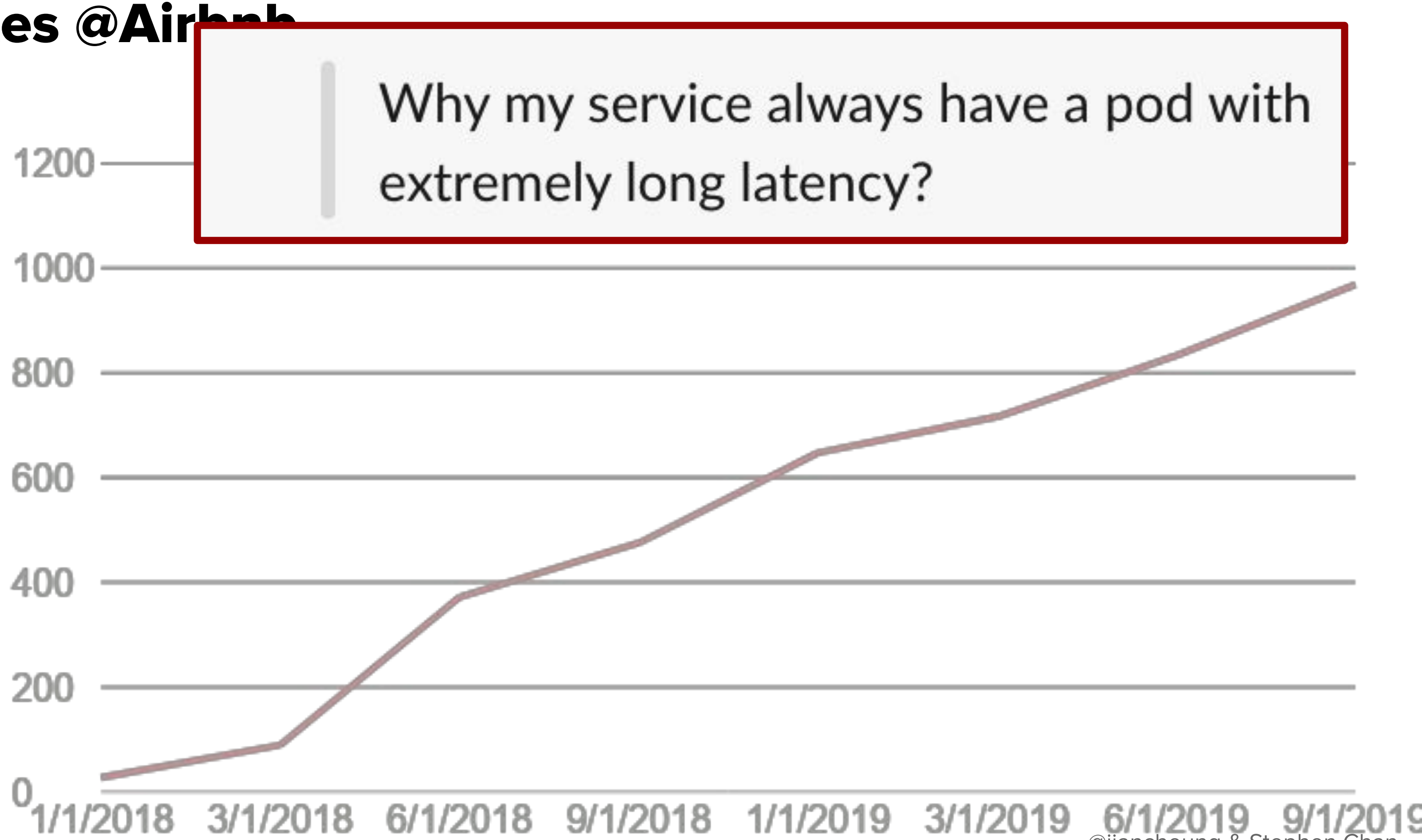


Lots of new services created in K8s & containers first.
But also a lot of migrations from EC2.

@jiancheung & Stephen Chan

Kubernetes @Airbnb

SERVICES

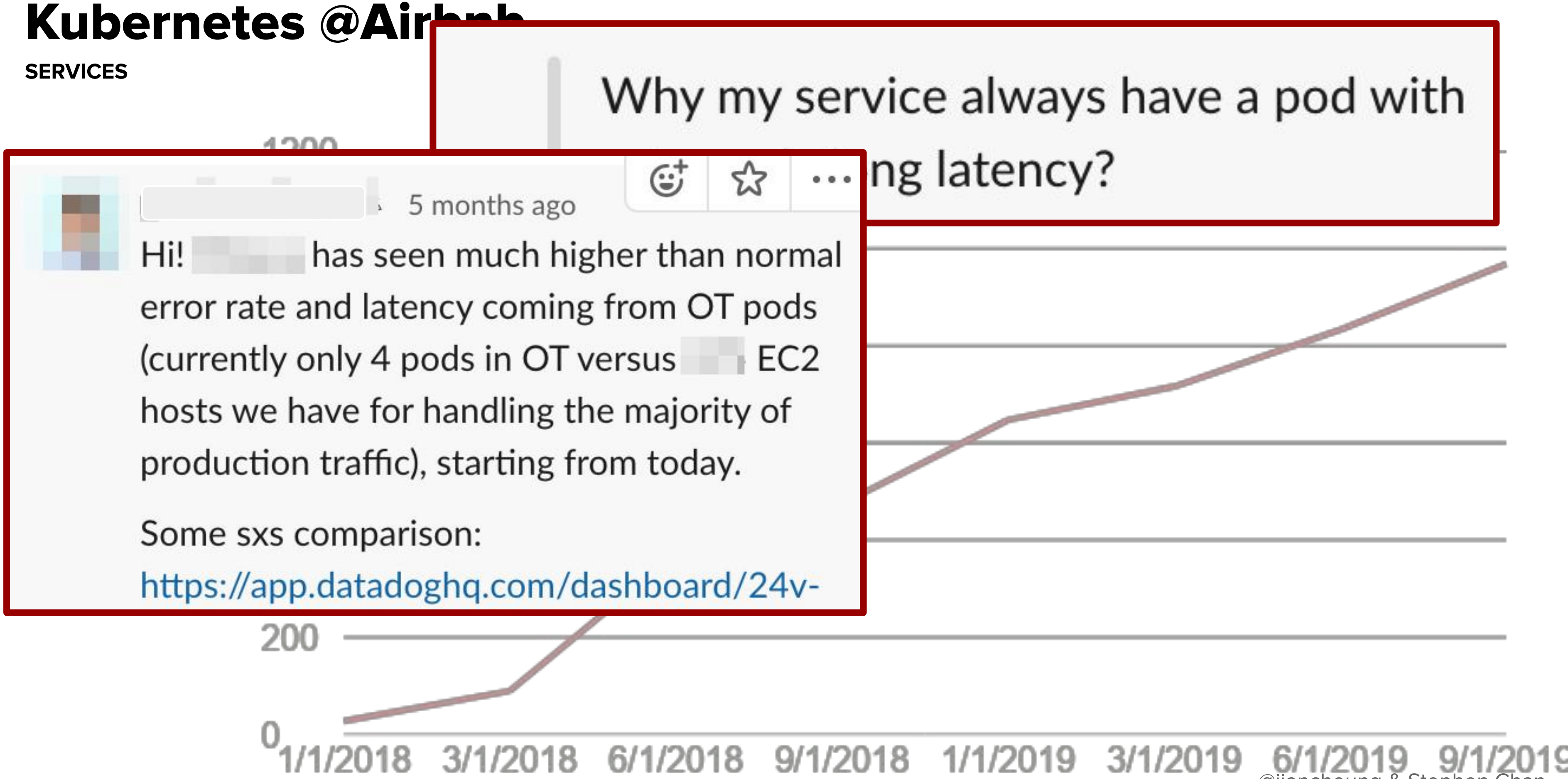


Why my service always have a pod with extremely long latency?

@jiancheung & Stephen Chan

Kubernetes @Airbnb

SERVICES



Why my service always have a pod with ng latency?

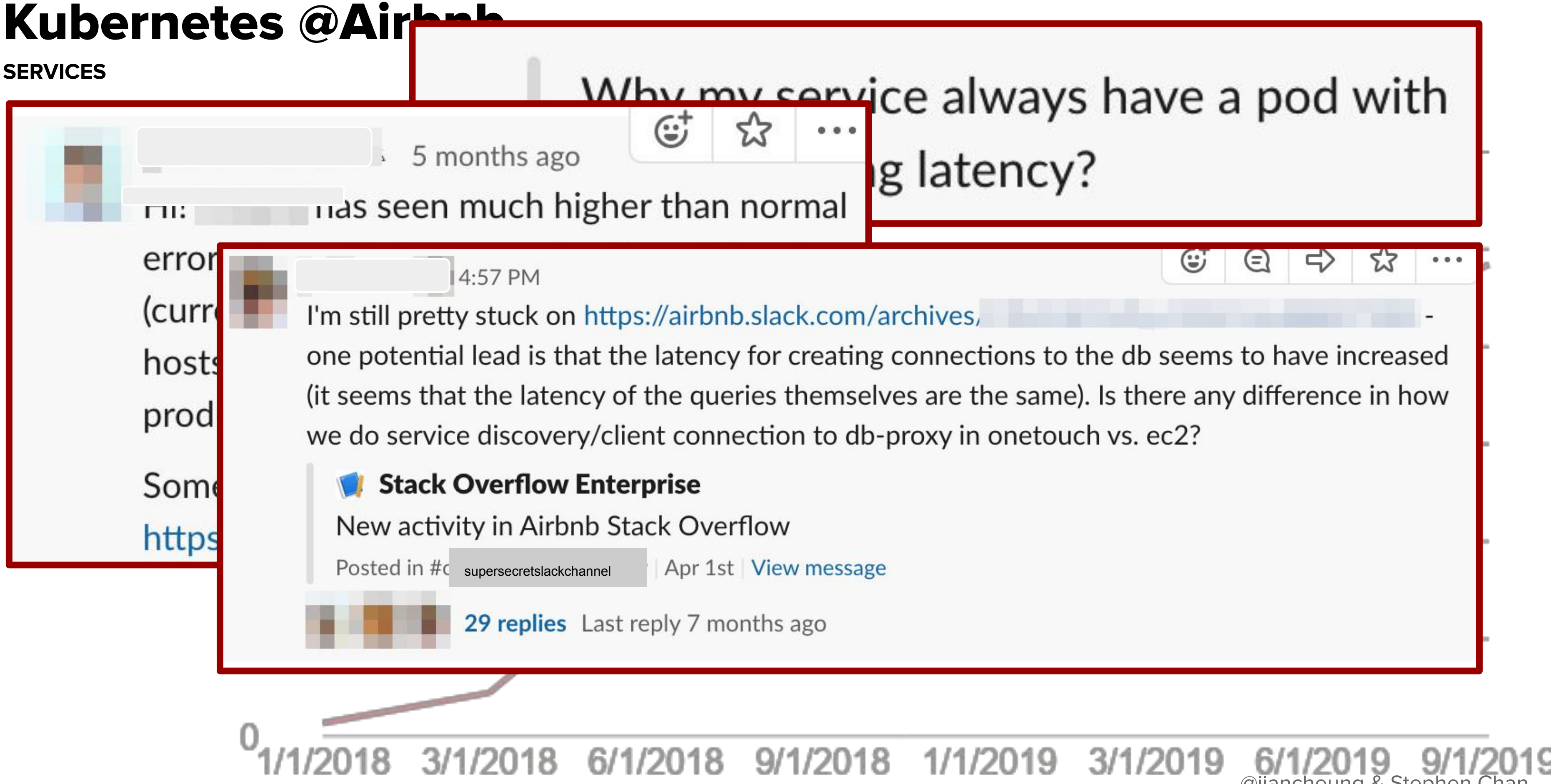
[redacted] 5 months ago

Hi! [redacted] has seen much higher than normal error rate and latency coming from OT pods (currently only 4 pods in OT versus [redacted] EC2 hosts we have for handling the majority of production traffic), starting from today.

Some sxs comparison:
https://app.datadoghq.com/dashboard/24v-

Kubernetes @Airbnb

SERVICES



Why my service always have a pod with
g latency?

[redacted] 5 months ago

Hi! [redacted] has seen much higher than normal

error

(curr

hosts

prod

Some

https

[redacted] 4:57 PM

I'm still pretty stuck on https://airbnb.slack.com/archives/[redacted] - one potential lead is that the latency for creating connections to the db seems to have increased (it seems that the latency of the queries themselves are the same). Is there any difference in how we do service discovery/client connection to db-proxy in onetouch vs. ec2?

Stack Overflow Enterprise

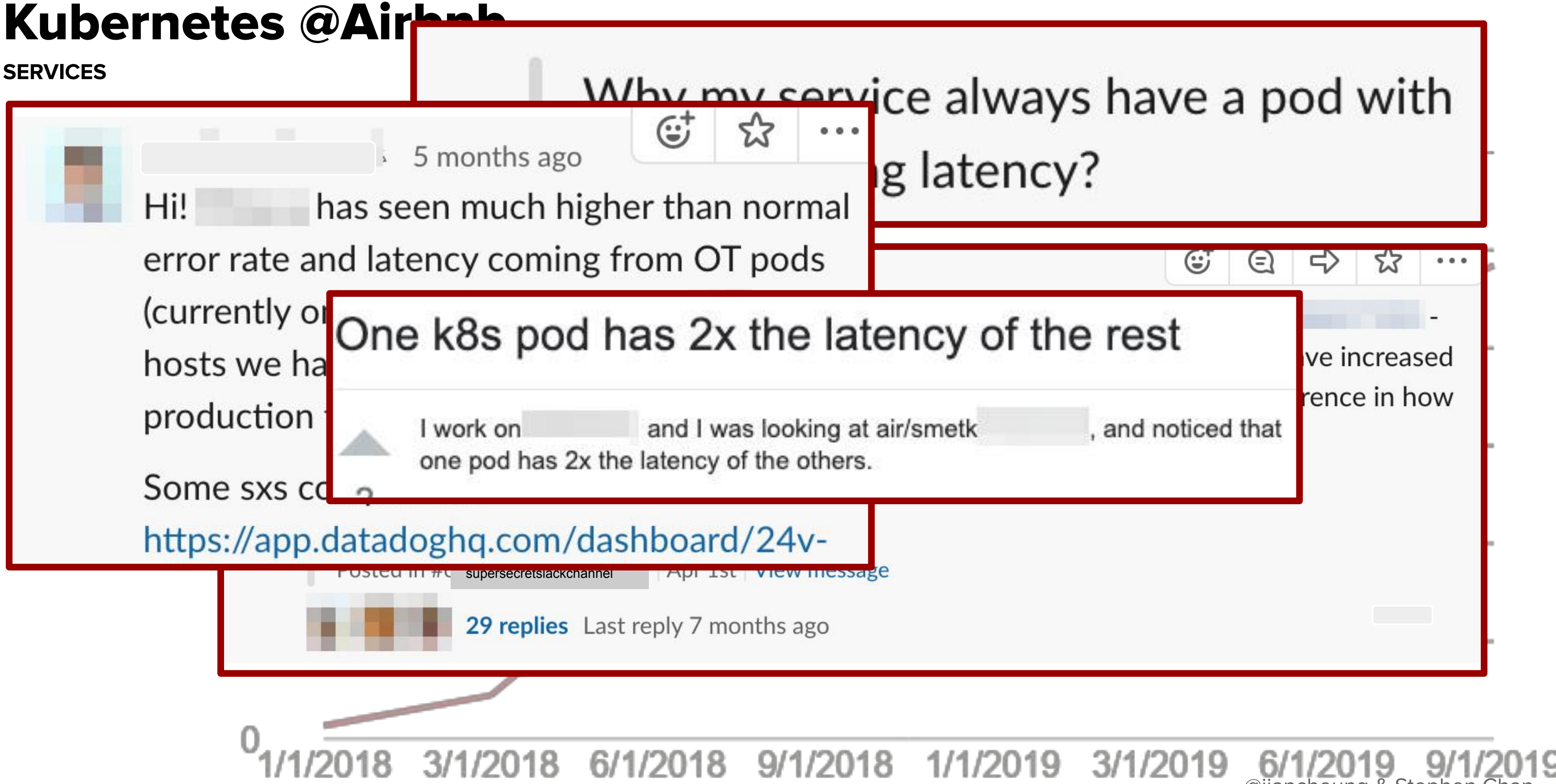
New activity in Airbnb Stack Overflow

Posted in #c supersecretslackchannel | Apr 1st | View message

29 replies Last reply 7 months ago

Kubernetes @Airbnb

SERVICES



Why my service always have a pod with … g latency?

[redacted] 5 months ago

Hi! [redacted] has seen much higher than normal error rate and latency coming from OT pods (currently o… hosts we ha… production …

Some sxs co…

https://app.datadoghq.com/dashboard/24v-

… ve increased … rence in how

Posted in #supersecretslackchannel | Apr 1st | View message

29 replies Last reply 7 months ago

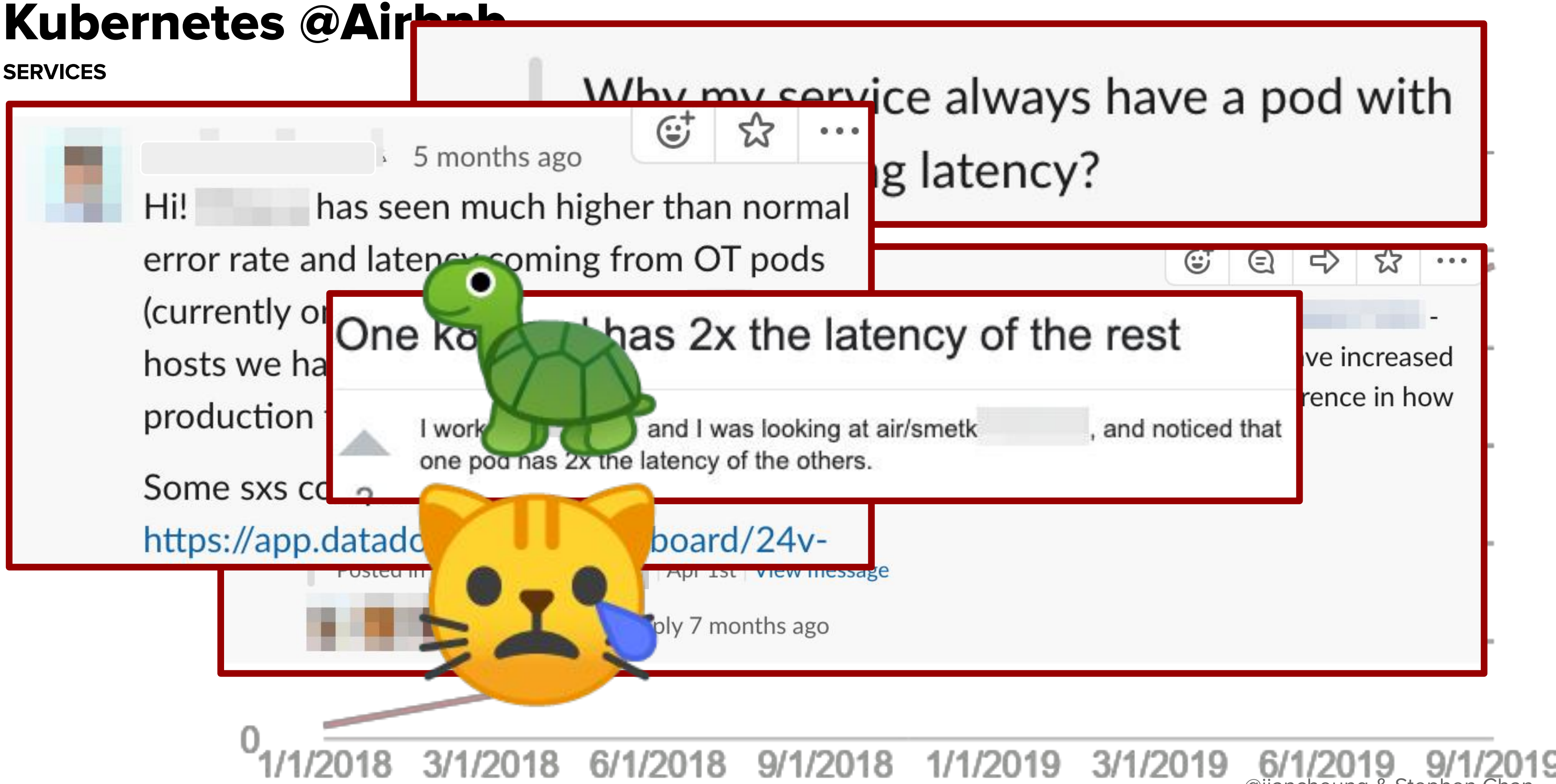
One k8s pod has 2x the latency of the rest

I work on [redacted] and I was looking at air/smetk [redacted], and noticed that one pod has 2x the latency of the others.

@jiancheung & Stephen Chan

Kubernetes @Airbnb

SERVICES



Why my service always have a pod with ...g latency?

[redacted] 5 months ago

Hi! [redacted] has seen much higher than normal error rate and latency coming from OT pods (currently o... hosts we ha... production ...

Some sxs co...

https://app.datado...board/24v-

One ka... has 2x the latency of the rest

I work ... and I was looking at air/smetk [redacted], and noticed that one pod has 2x the latency of the others.

[redacted] -

...ve increased ...rence in how

Posted in ... Apr 1st | View message

...ply 7 months ago

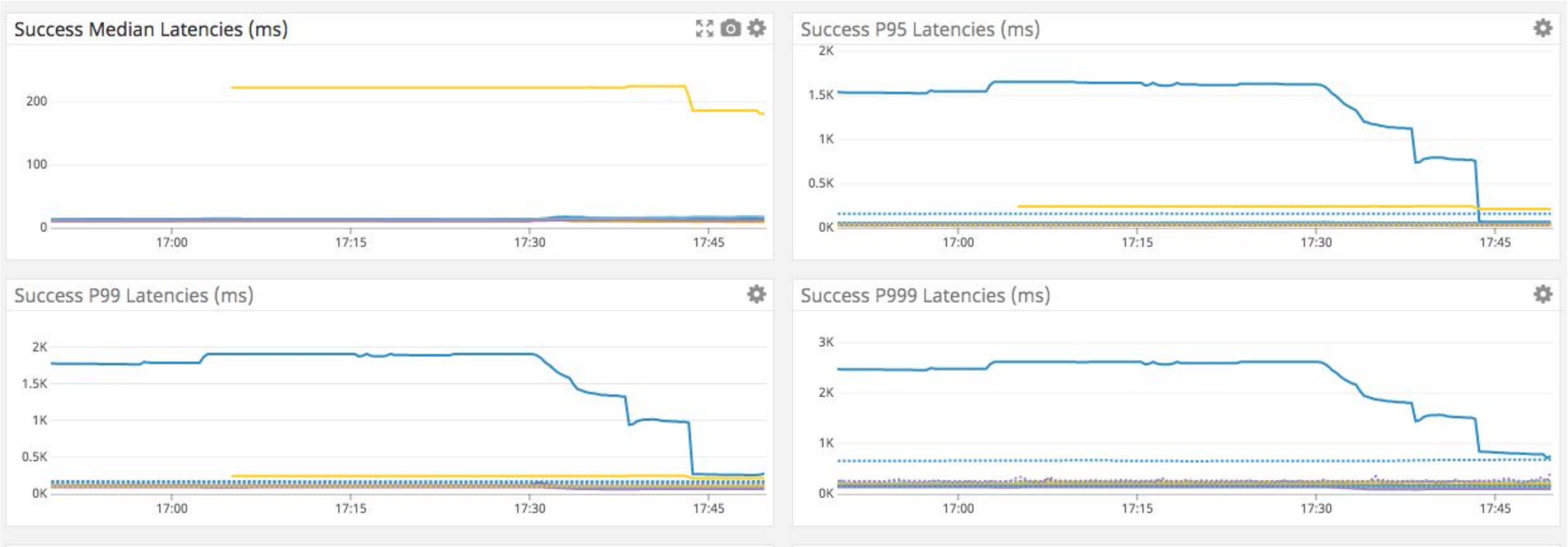
@jiancheung & Stephen Chan

So let’s dive in

Latencies *Improved*?

⏱️ 📉

Latencies *Improved?*



@jiancheung & Stephen Chan

Latencies *Improved?*

- Migrated from ec2 to Kubernetes (+ containers)
- No code changes.
- Same amount of CPU / memory
- Java service
- Latencies dramatically improved
- Spun up early 2018

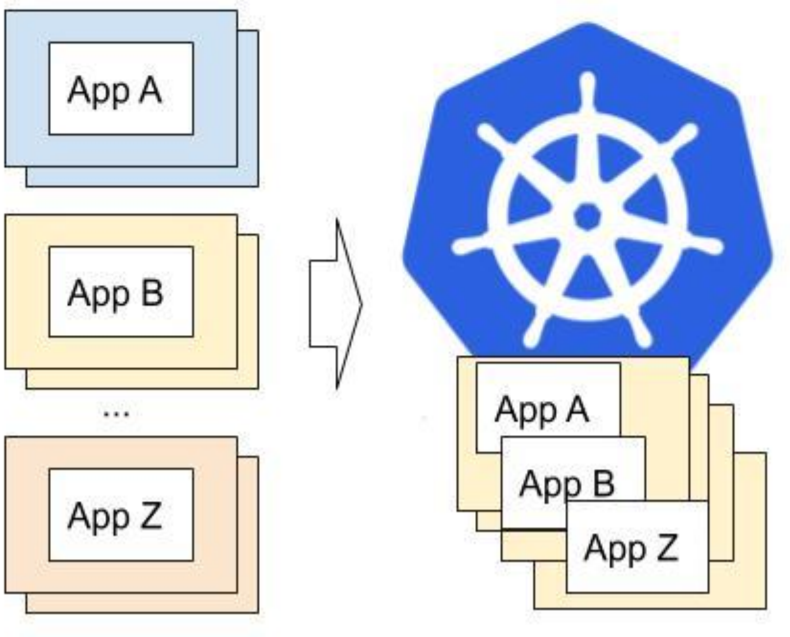
Latencies *Improved?*

- Migrated from ec2 to Kubernetes (+ containers)
- No code changes.
- Same amount of CPU / memory
- Java service
- Latencies dramatically improved
- **Spun up early 2018** 🤔

Latencies *Improved?*

The service was running on previous generation's hardware. The migration just so happened to have also upgraded the service's hardware.

> "Just wow. It's a better box with faster network i/o that's cheaper"



Latencies *Improved?*

Did Kubernetes make my p95s ~~worse~~ better?

Latencies *Improved?*

Did Kubernetes make my p95s ~~worse~~ better?

NO (but we actually tell our customers yes)

Latencies *Improved?*

Did Kubernetes make my p95s ~~worse~~ better?

NO (but we actually tell our customers yes)

… because hardware choices have to be made anyways and

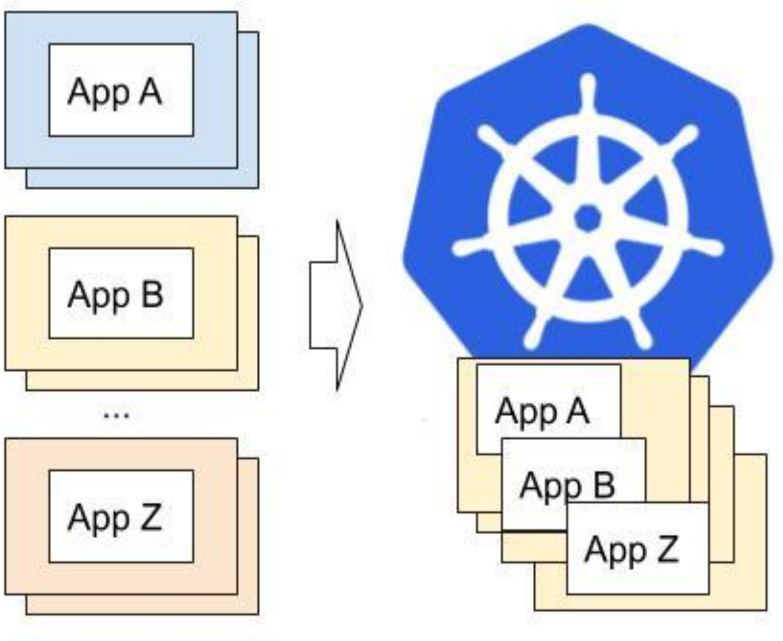
usually instance types aren’t intentionally picked to match app

Latencies *Improved?*

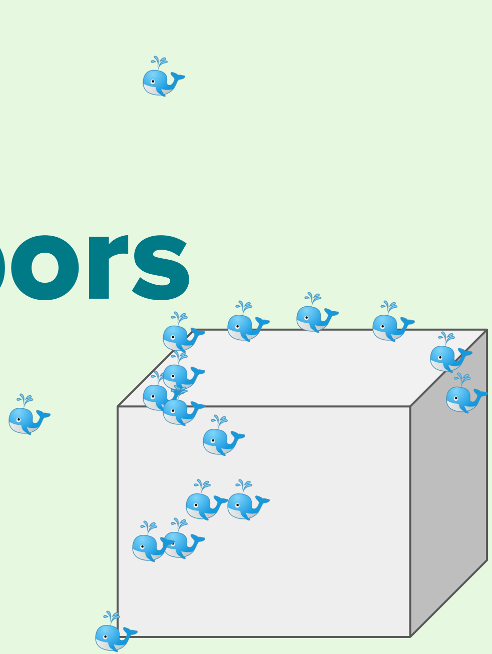
Lesson:

The “things” the app is running on can be different for the ***better*** and ***worse***.

- Hardware
- Host OS



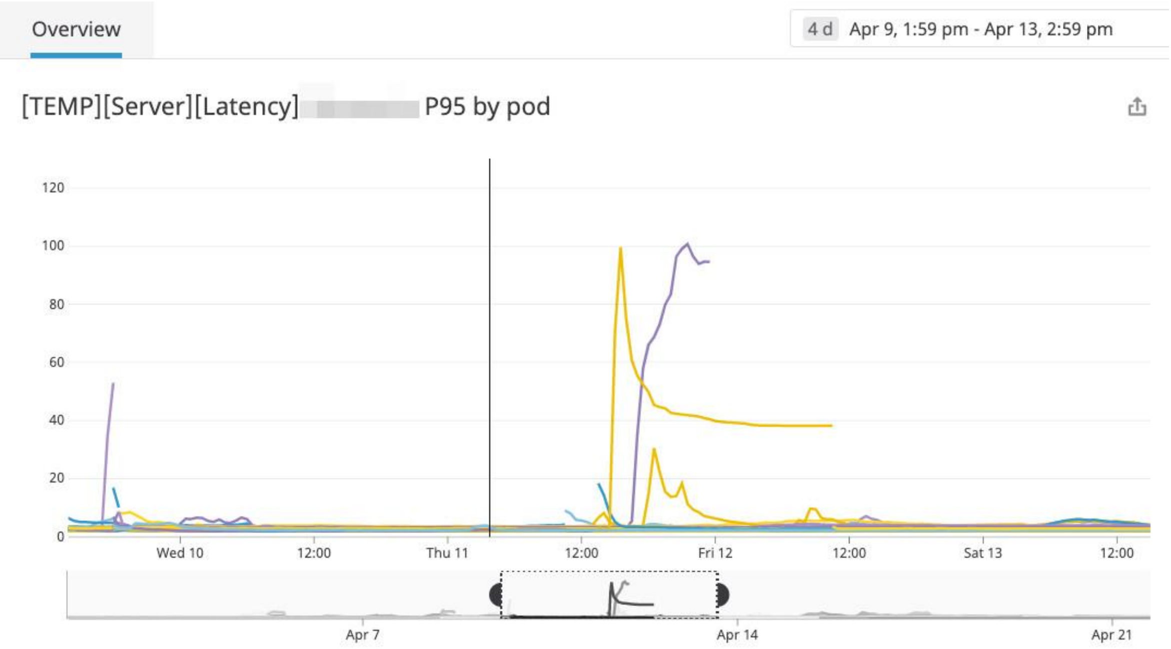
Noisy Neighbors



@jiancheung & Stephen Chan

Noisy Neighbors

Sometimes it’s just certain pods



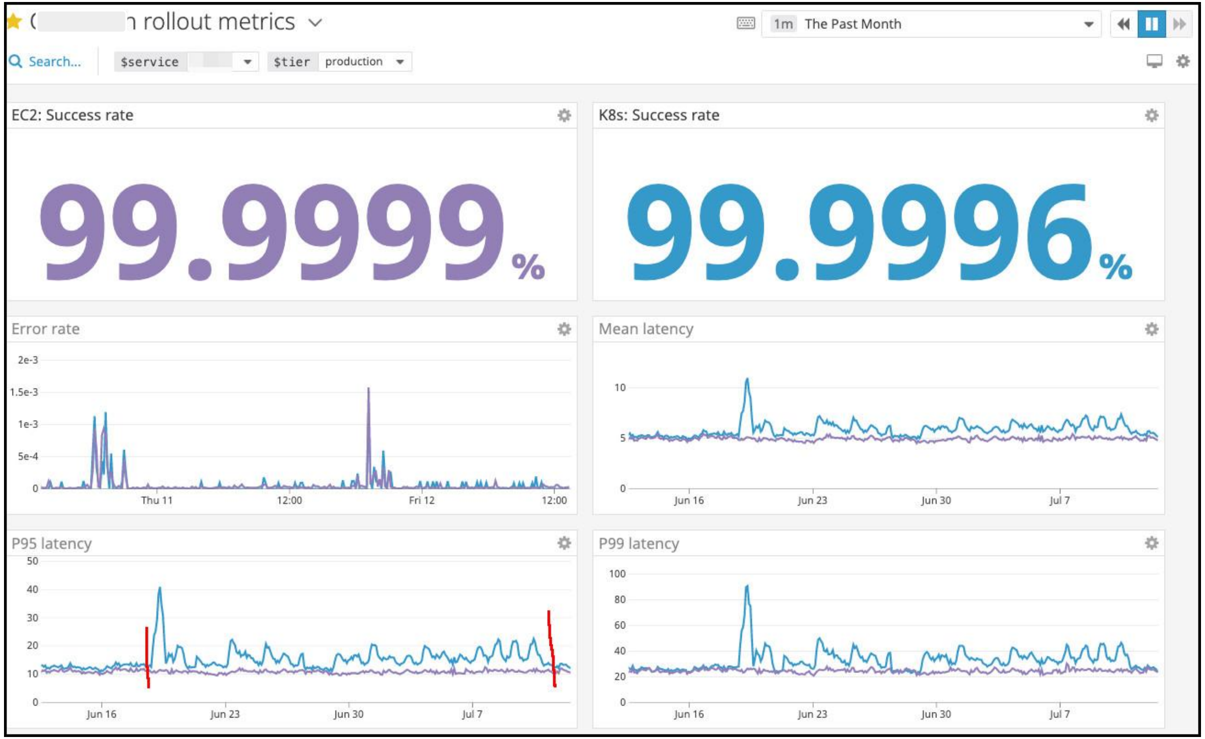
Overview

4 d Apr 9, 1:59 pm - Apr 13, 2:59 pm

[TEMP][Server][Latency] [redacted] P95 by pod

Noisy Neighbors

Sometimes it’s becomes constant



([redacted] n rollout metrics

1m The Past Month

Search...

$service [redacted]

$tier production

EC2: Success rate

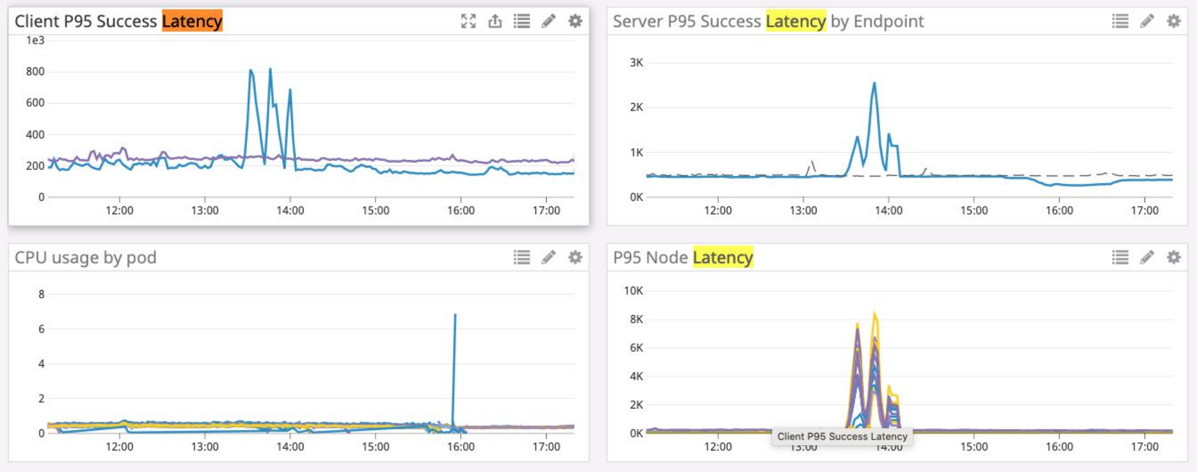
99.9999%

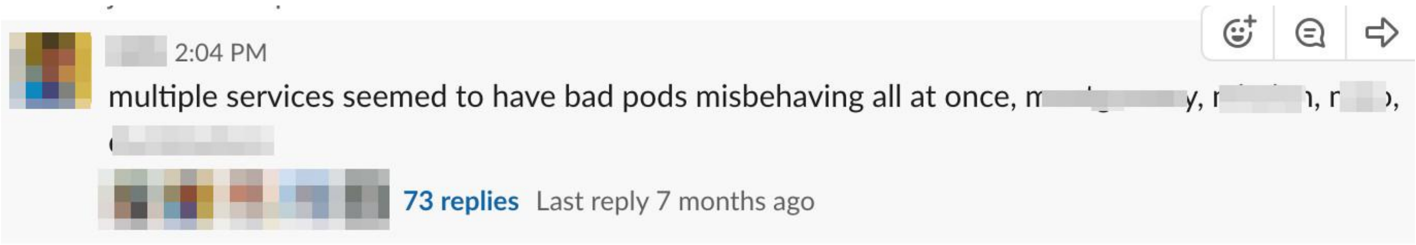
K8s: Success rate

99.9996%

Noisy Neighbors

Sometimes it’s an incident





[redacted] 2:04 PM

multiple services seemed to have bad pods misbehaving all at once, m[redacted]y, r[redacted]n, r[redacted],
([redacted]

73 replies Last reply 7 months ago

@jiancheung & Stephen Chan

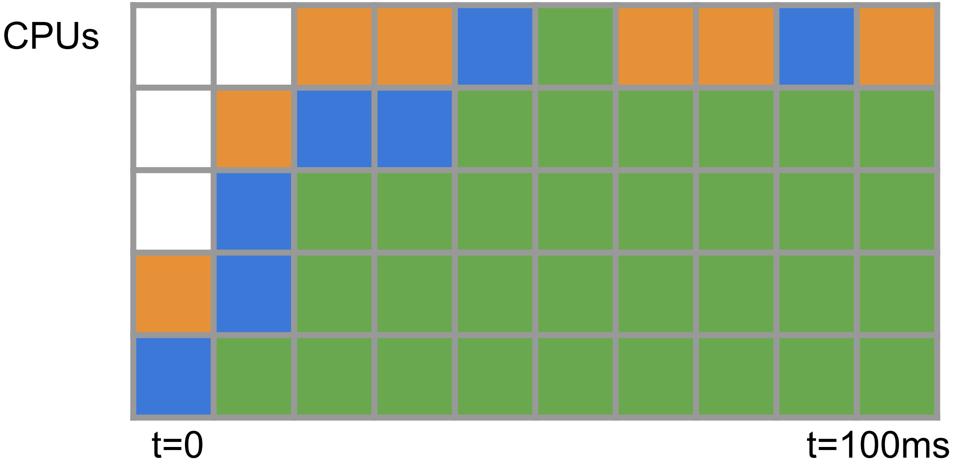
Noisy Neighbors

So what happened? (hint: it’s in the title)

Noisy Neighbors

Multiple containers/apps/processes sharing the resources of one computer.

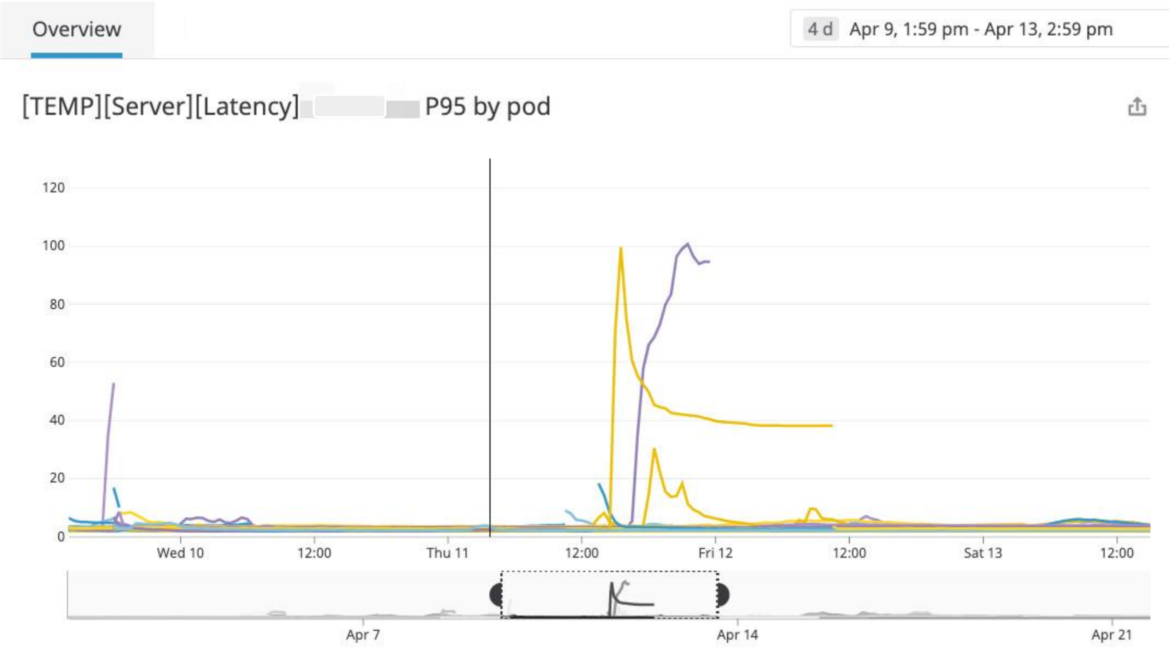
There's only so much CPU to go around.



Noisy Neighbors

Sometimes it’s just certain pods

Specifically pods that were co-located with a **Service Kale**



Overview

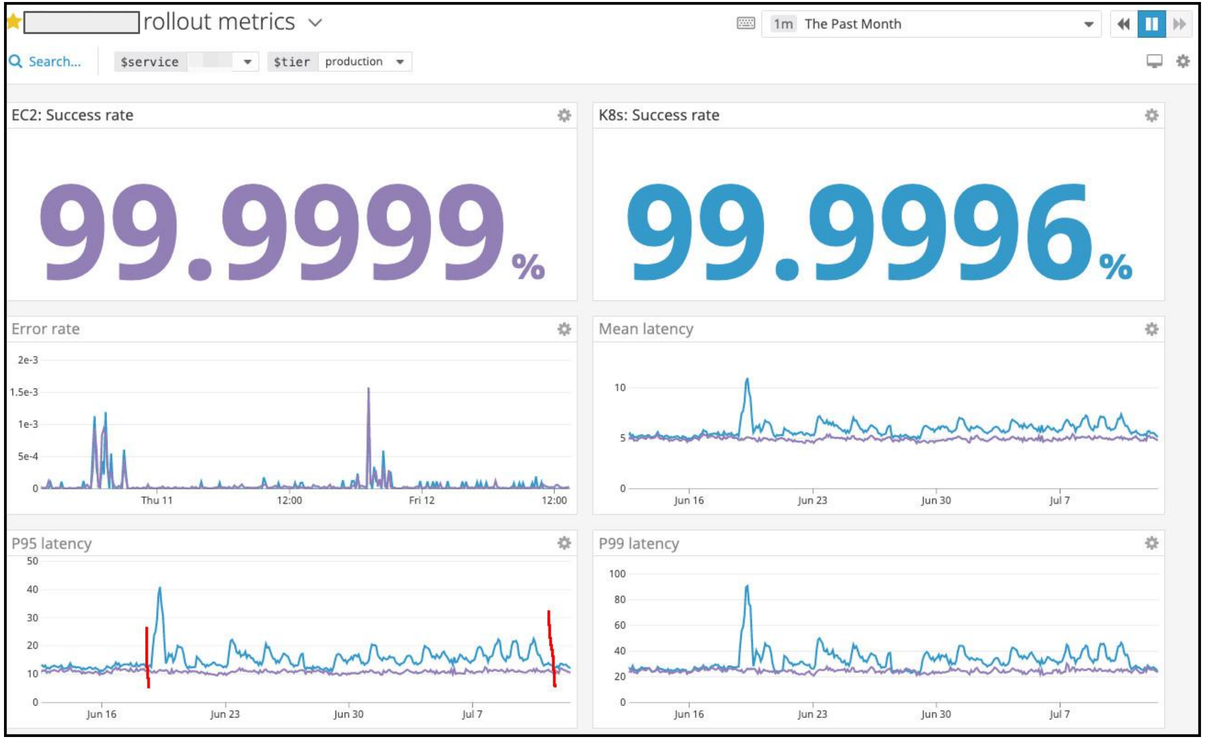
4 d Apr 9, 1:59 pm - Apr 13, 2:59 pm

[TEMP][Server][Latency] [redacted] P95 by pod

Noisy Neighbors

Sometimes it’s becomes constant

This happened when **Service Kale** migrated to the same cluster



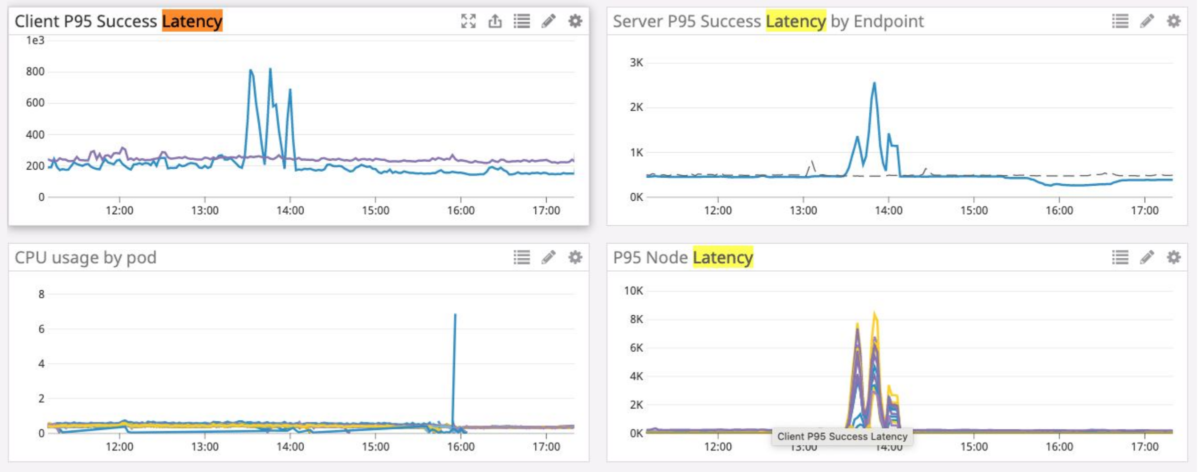
rollout metrics

1m The Past Month

$service

$tier production

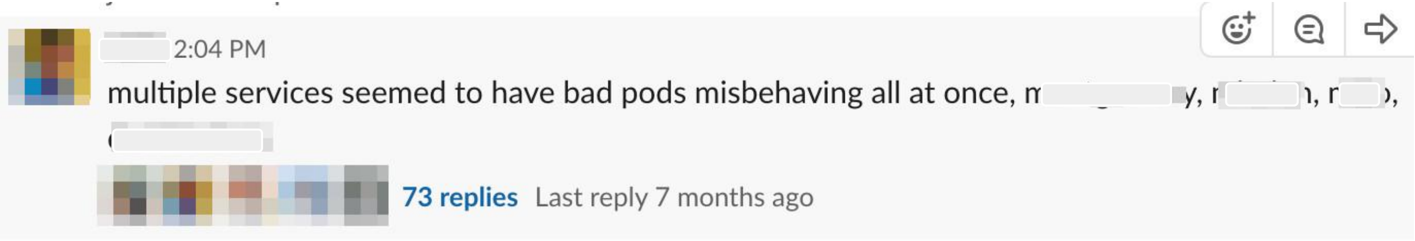
Noisy Neighbors



Sometimes it’s an incident

Okay this time not **Service Kale**.

This happened when a staging service accidentally got deployed to the wrong cluster



[redacted] 2:04 PM

multiple services seemed to have bad pods misbehaving all at once, m[redacted]y, r[redacted]n, r[redacted]),
([redacted]

73 replies Last reply 7 months ago

Noisy Neighbors

```
Limits:
  memory:  7Gi
Requests:
  cpu:     1500m
  memory:  7Gi
```

In the early days of Airbnb & Kubernetes,
we decided not to set CPU limits because it had seemed to hurt performance 🙃

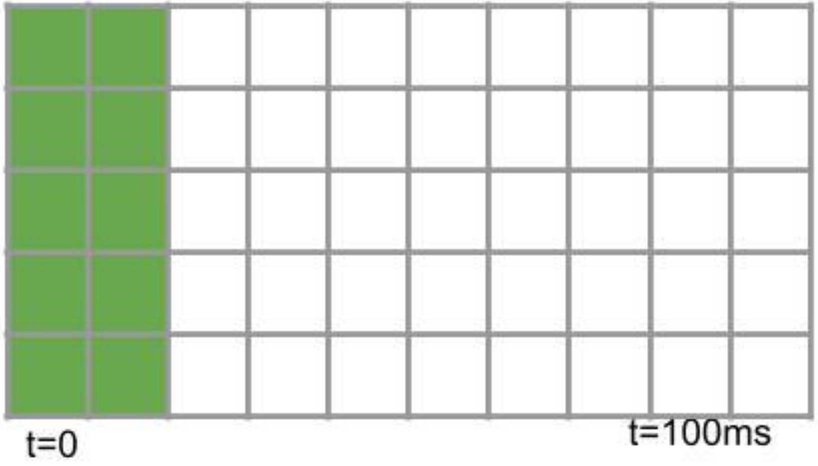
Noisy Neighbors

```
Limits:
  cpu:     1500m # <- important
  memory:  7Gi
Requests:
  cpu:     1500m
  memory:  7Gi
```

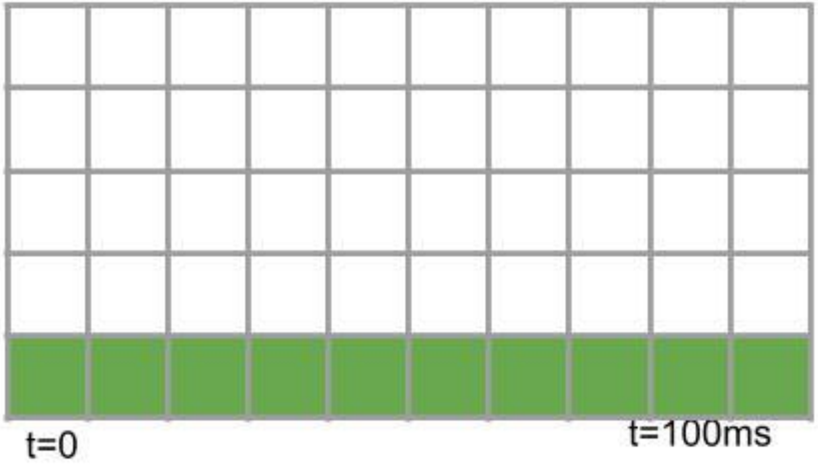
Easy and simple right?

Noisy Neighbors

```
Limits:
  cpu:     10m
Requests:
  cpu:     10m
```



How do you spend your 10 cpu.quota?



Given a CPU CFS quota of 100ms,
if you use it all up in the first 20ms, then you get throttled for 80ms.

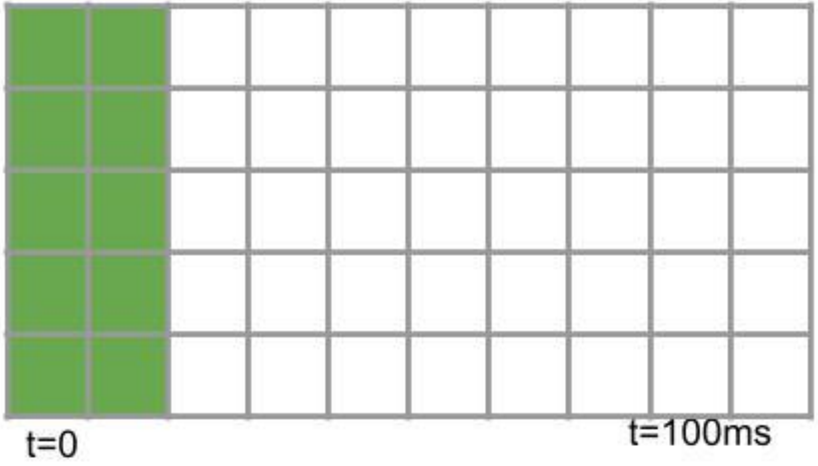
Noisy Neighbors

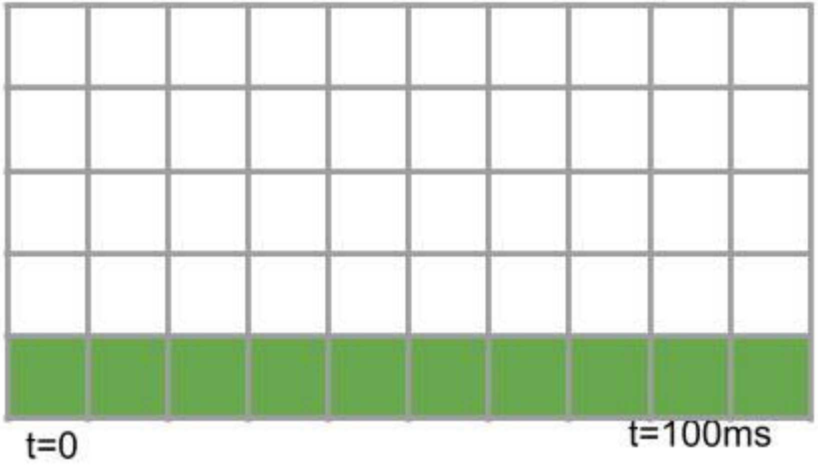
Both these cases, our metric collectors would have show similar/low CPU utilization.

This makes fine-grained hotspots hard to detect.

Things we've tried:

1. Changing CFSQuota (didn't help for our cases)
2. Finer grain CPU metrics collector + more CPU allocation
3. Set CPU limits





@jiancheung & Stephen Chan

Noisy Neighbors

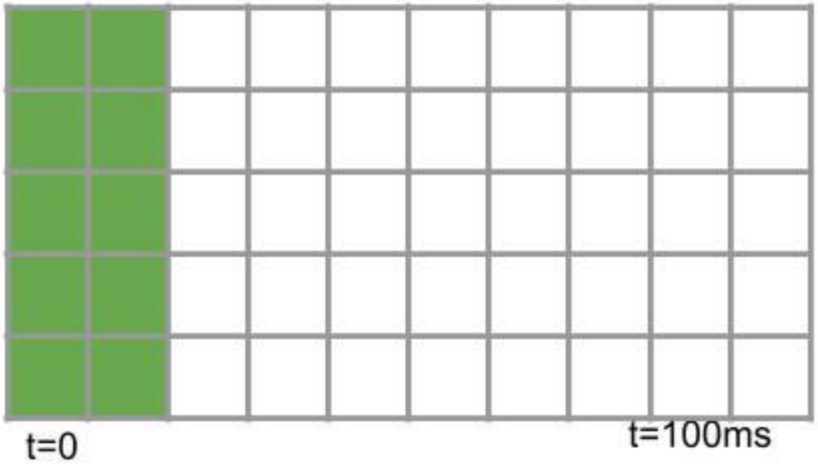
Things we want to try:

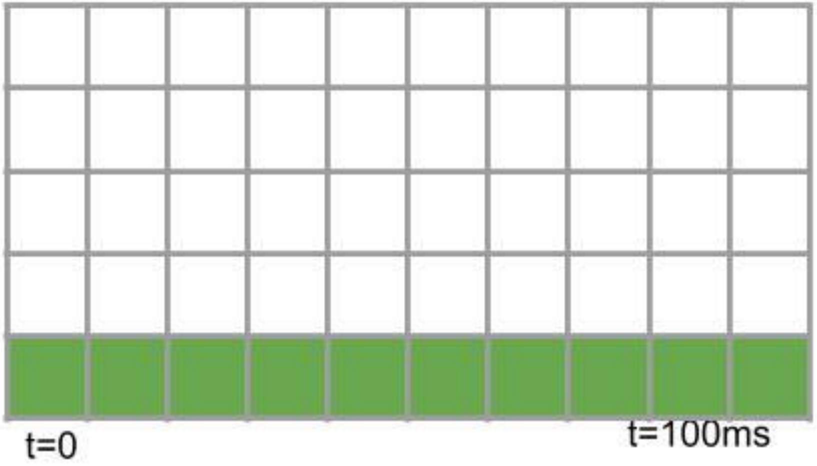
- CPU pinning / CPU sets

(Avoid setting CPU limits for Guaranteed pods)
https://github.com/kubernetes/kubernetes/issues/51135

(Disable cpu quota(use only cpuset) for pod Guaranteed)
https://github.com/kubernetes/kubernetes/issues/70585

(Unset CPU CFS quota when CPU sets are in use)
https://github.com/kubernetes/kubernetes/pull/75682





Noisy Neighbors

Did Kubernetes make my p95s worse?

Noisy Neighbors

Did Kubernetes make my p95s worse?

YES

Noisy Neighbors

Did Kubernetes make my p95s worse?

YES

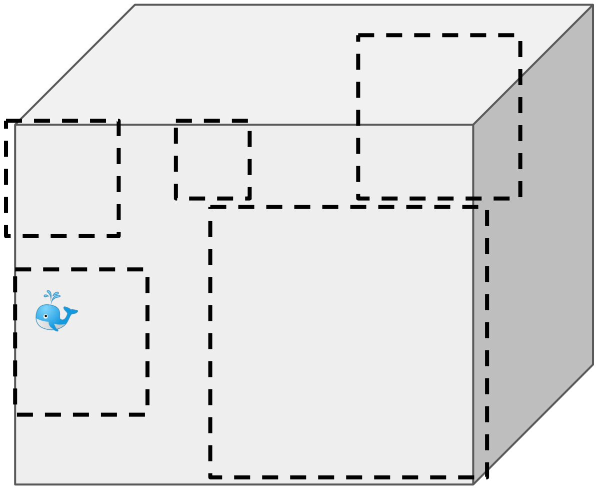
Multitenancy is awesome but it’s hard to not take *some* performance hits from it.

Noisy Neighbors

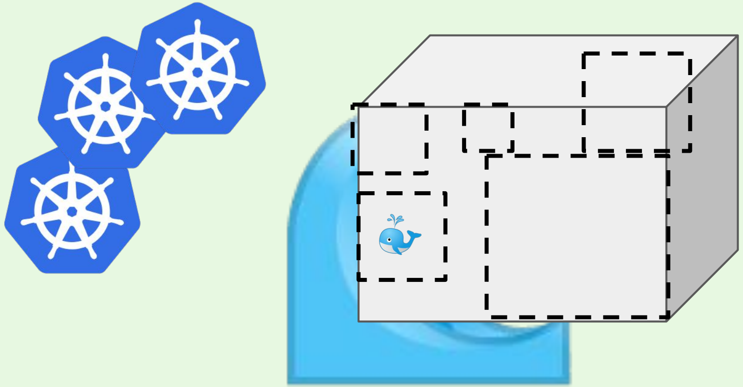
Lesson:

Containers should be contained.

Set resource limits.



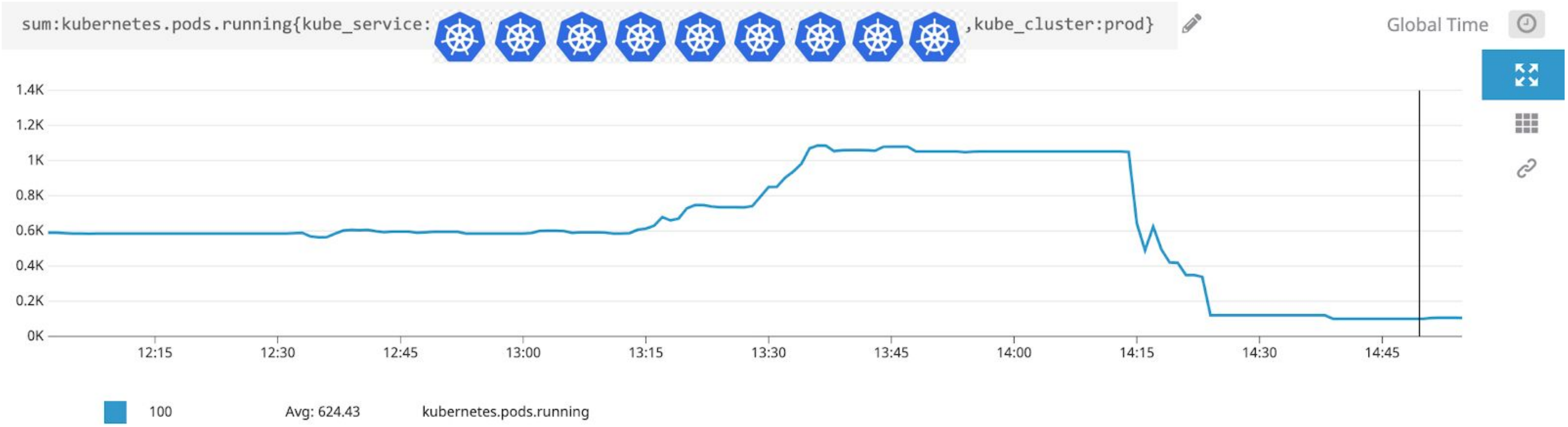
Noisy Neighbors, made worse by Kubernetes



@jiancheung & Stephen Chan

Noisy Neighbors, made worse by Kubernetes

When autoscaling goes up and to the right...

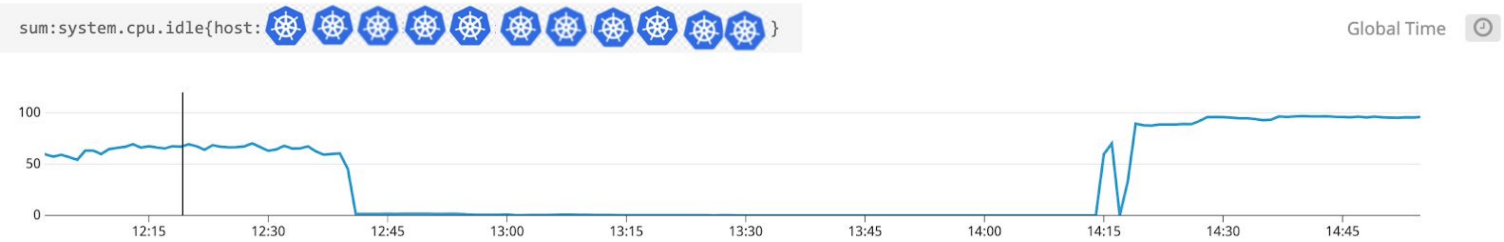


```
sum:kubernetes.pods.running{kube_service:[redacted],kube_cluster:prod}
```

🟦 100 Avg: 624.43 kubernetes.pods.running

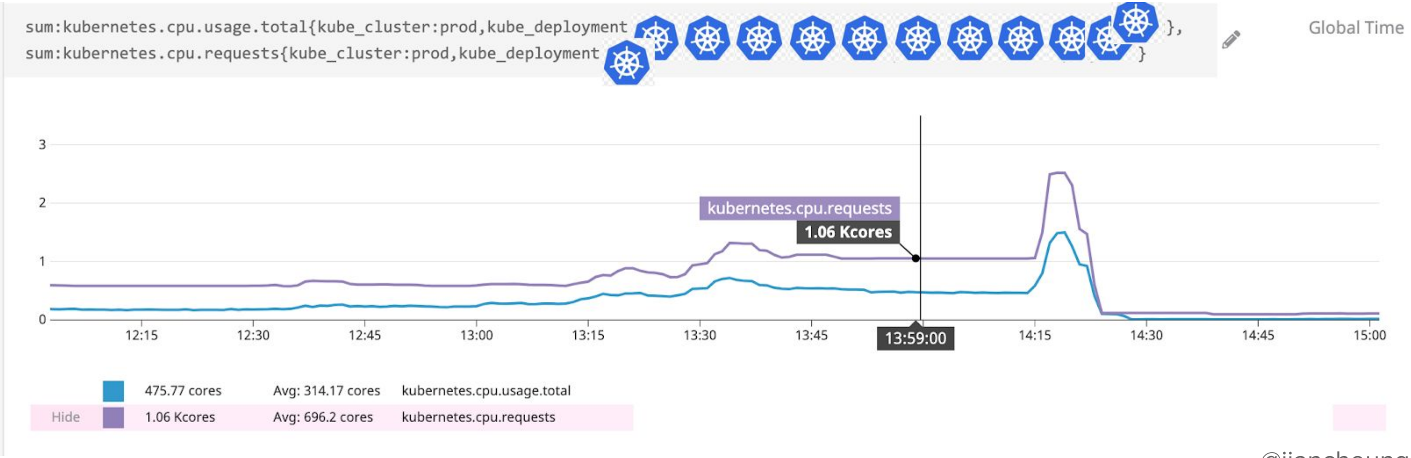
Noisy Neighbors, made worse by Kubernetes

1 host is starved...



```
sum:system.cpu.idle{host: }
```

But in aggregate load is fine



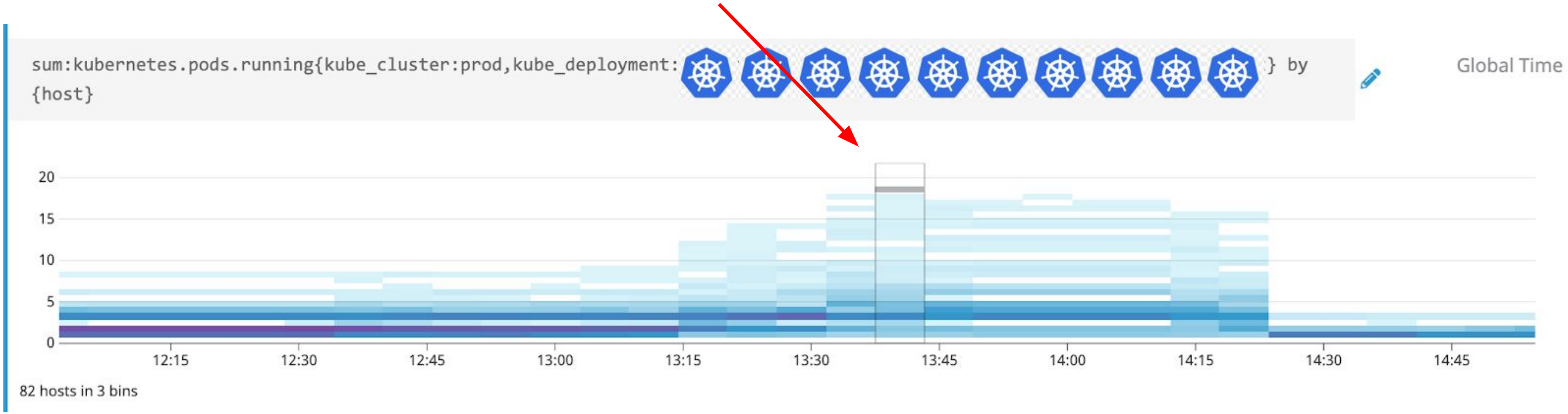
```
sum:kubernetes.cpu.usage.total{kube_cluster:prod,kube_deployment },
sum:kubernetes.cpu.requests{kube_cluster:prod,kube_deployment }
```

🟦	475.77 cores	Avg: 314.17 cores	kubernetes.cpu.usage.total
🟪	1.06 Kcores	Avg: 696.2 cores	kubernetes.cpu.requests

@jiancheung & Stephen Chan

Noisy Neighbors, made worse by Kubernetes

18 identical service pods running
on a single host??



```
sum:kubernetes.pods.running{kube_cluster:prod,kube_deployment:[redacted]} by
{host}
```

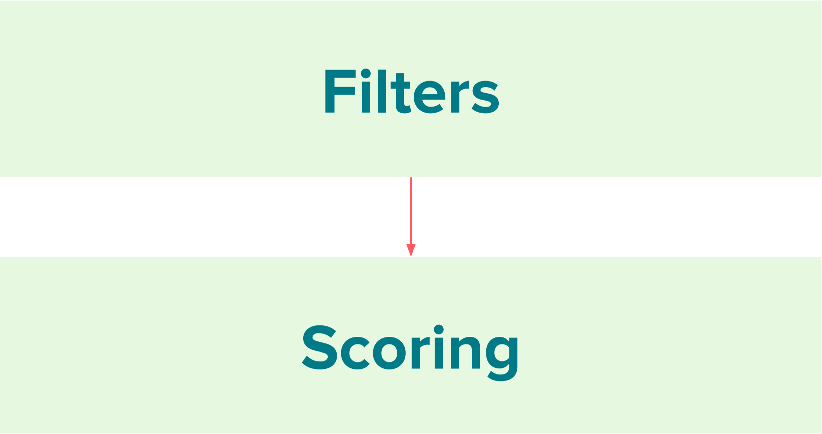
Global Time

82 hosts in 3 bins

Noisy Neighbors, made worse by Kubernetes

Scheduling primer

Where MUST or MUST NOT my pod run?



Filters

Scoring

Some filters:

- Resource
- Topology
- Required affinity

Where SHOULD or SHOULD NOT my pod run?

Some scoring priorities:

- Preferred affinity
- Spreading by topology
- **Image locality**

Noisy Neighbors, made worse by Kubernetes

Did Kubernetes make my p95s worse?

Noisy Neighbors, made worse by Kubernetes

Did Kubernetes make my p95s worse?

YES

Noisy Neighbors, made worse by Kubernetes

Did Kubernetes make my p95s worse?

YES

The scheduler can even work against you in pathological cases.

Pod Topology Spread Constraints might help avoid this (but we haven’t tried yet)

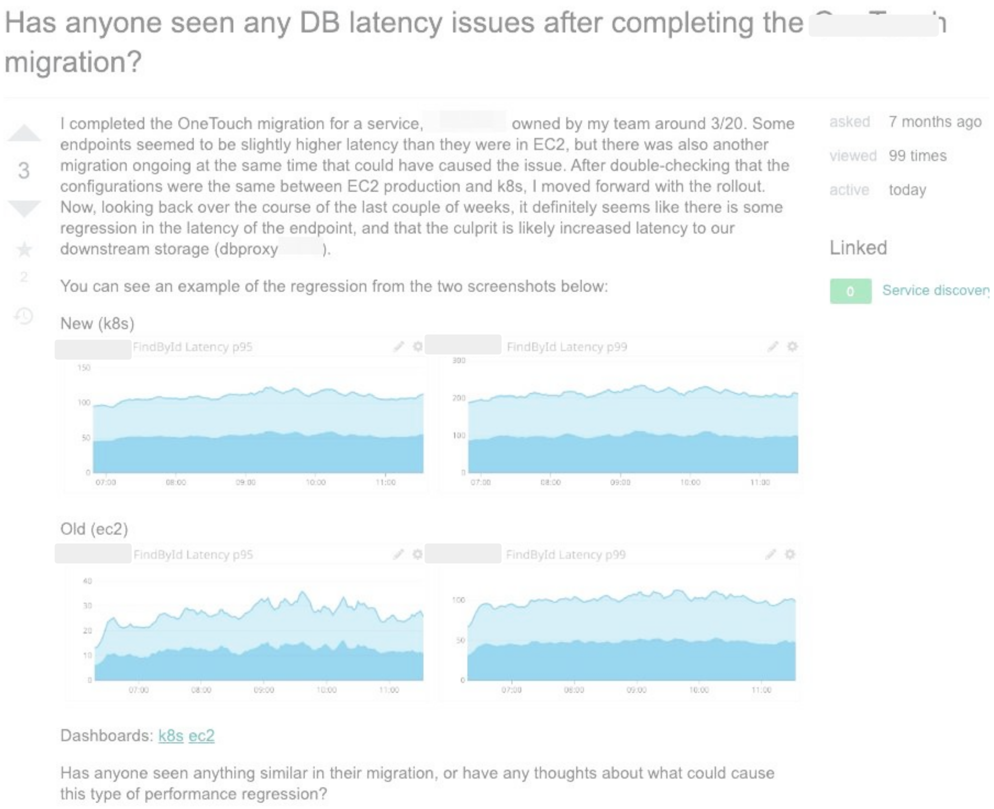
Noisy Neighbors, made worse by Kubernetes

Lessons:

- K8s services can cause traffic imbalance (especially when using iptables proxier)
- Autoscaling v1 uses average CPU across all pods; this can cause pathological behavior

Write Once, Run Anywhere

Write Once, Run Anywhere



Has anyone seen any DB latency issues after completing the [redacted] migration?

3

2

I completed the OneTouch migration for a service, [redacted] owned by my team around 3/20. Some endpoints seemed to be slightly higher latency than they were in EC2, but there was also another migration ongoing at the same time that could have caused the issue. After double-checking that the configurations were the same between EC2 production and k8s, I moved forward with the rollout. Now, looking back over the course of the last couple of weeks, it definitely seems like there is some regression in the latency of the endpoint, and that the culprit is likely increased latency to our downstream storage (dbproxy [redacted]).

You can see an example of the regression from the two screenshots below:

New (k8s)

Old (ec2)

Dashboards: k8s ec2

Has anyone seen anything similar in their migration, or have any thoughts about what could cause this type of performance regression?

asked	7 months ago
viewed	99 times
active	today

Linked

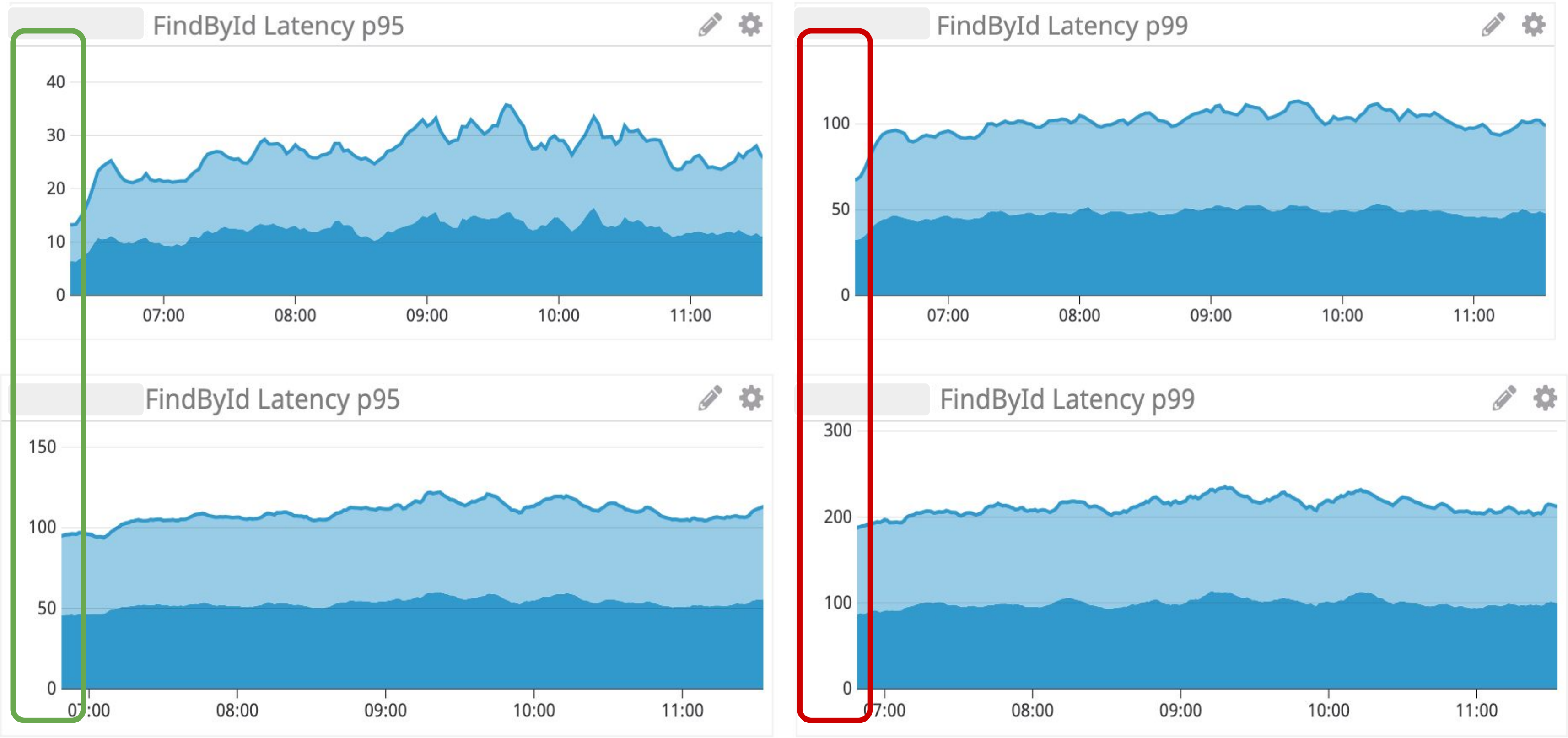
0 Service discover

Key points are:

- Java application
- P95 latencies 30ms -> 100ms
- P99 latencies 100ms -> 200ms
- Specifically DB connections

@jiancheung & Stephen Chan

Write Once, Run Anywhere



@jiancheung & Stephen Chan

Write Once, Run Anywhere

For a specific endpoint, we had created a new threadpool **per** request.

Can be fixed by reusing a threadpool in a static context.

Write Once, Run Anywhere

For a specific endpoint, we had created a new threadpool **per** request.

Can be fixed by reusing a threadpool in a static context.

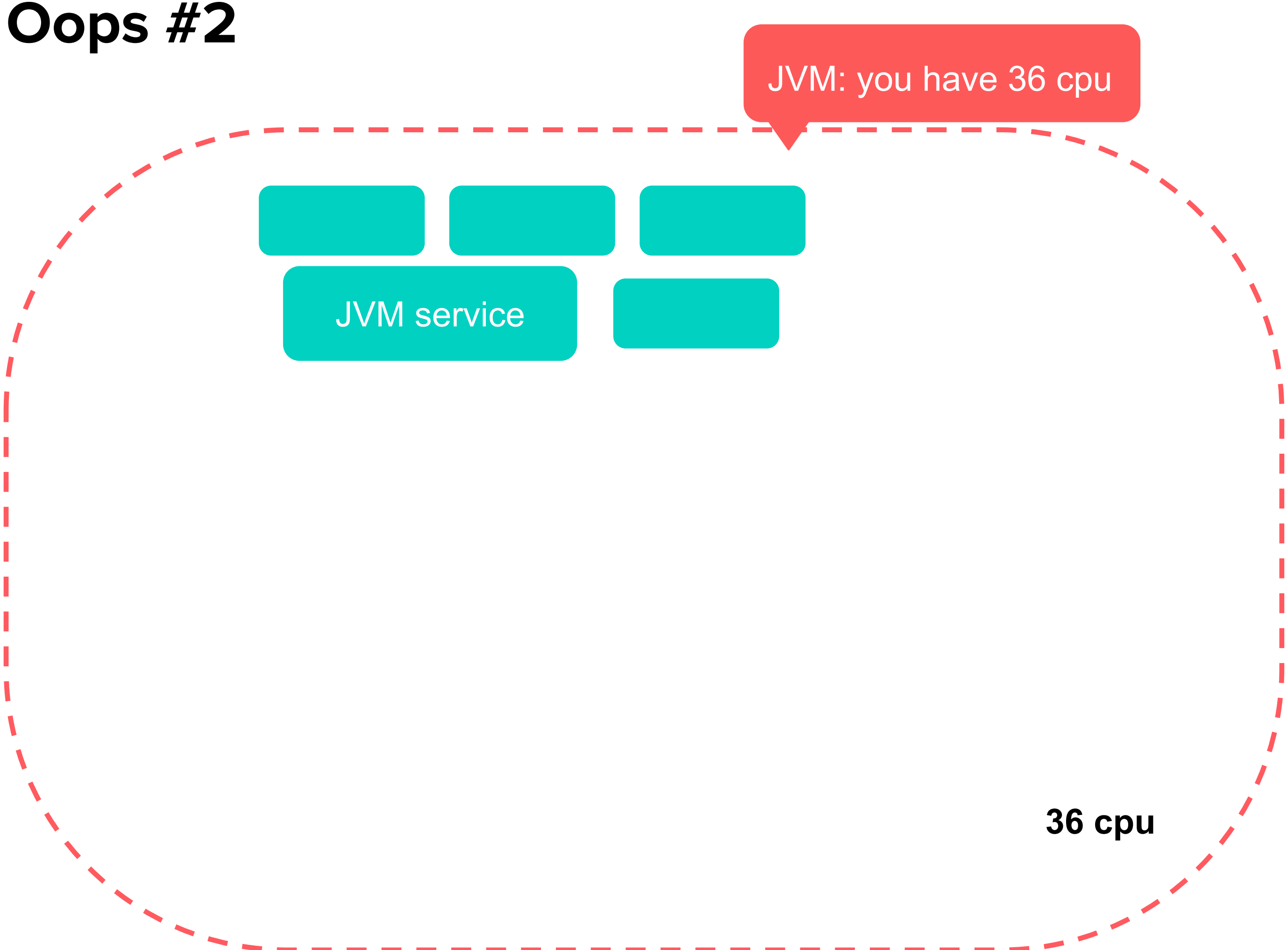
But why did this work before kubernetes?

Oops #2

Pod

Container / Application

Node

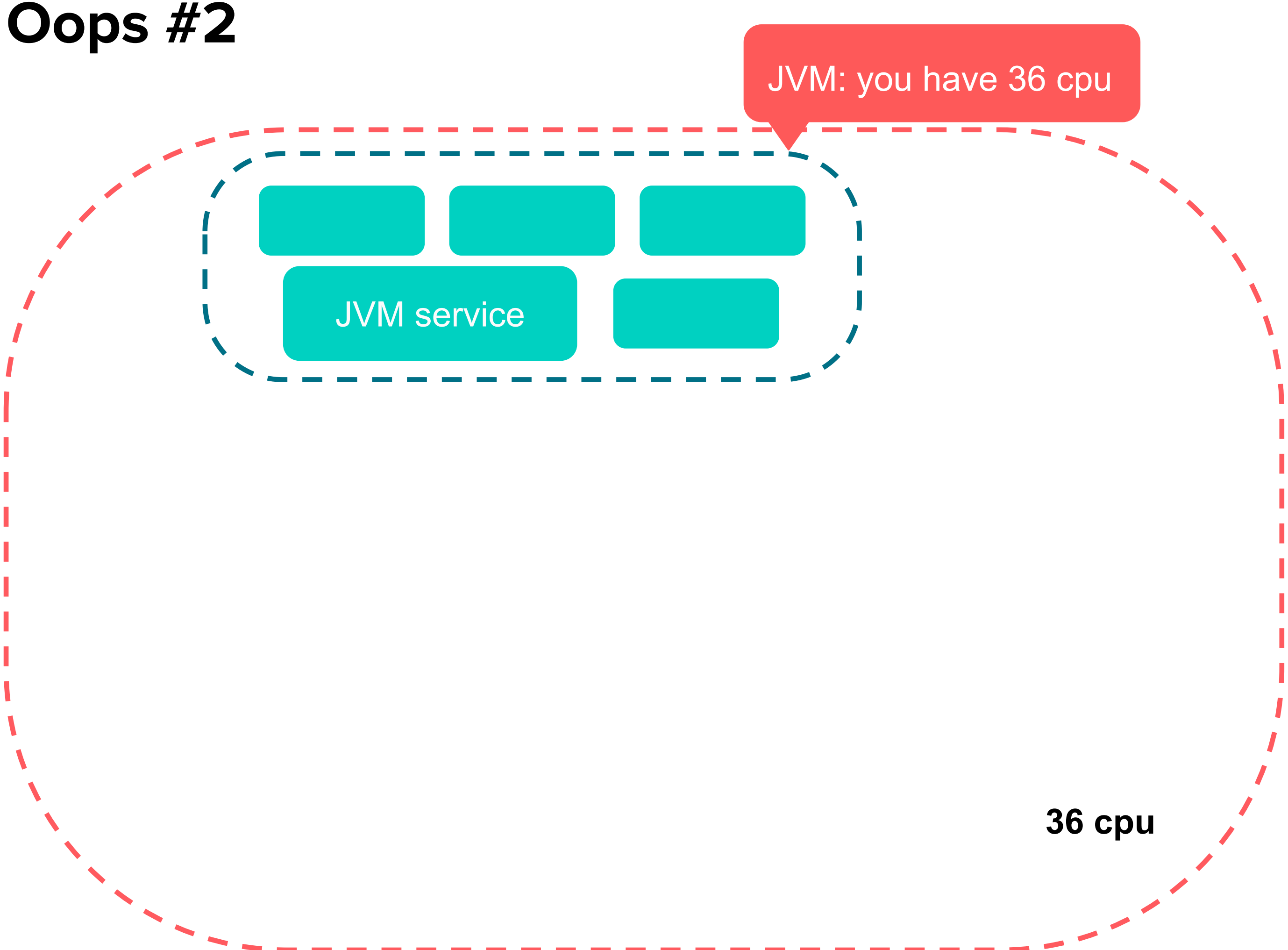


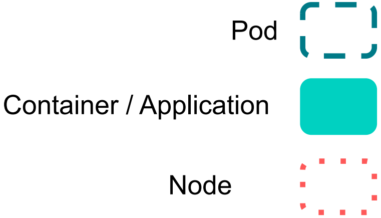
JVM: you have 36 cpu

JVM service

36 cpu

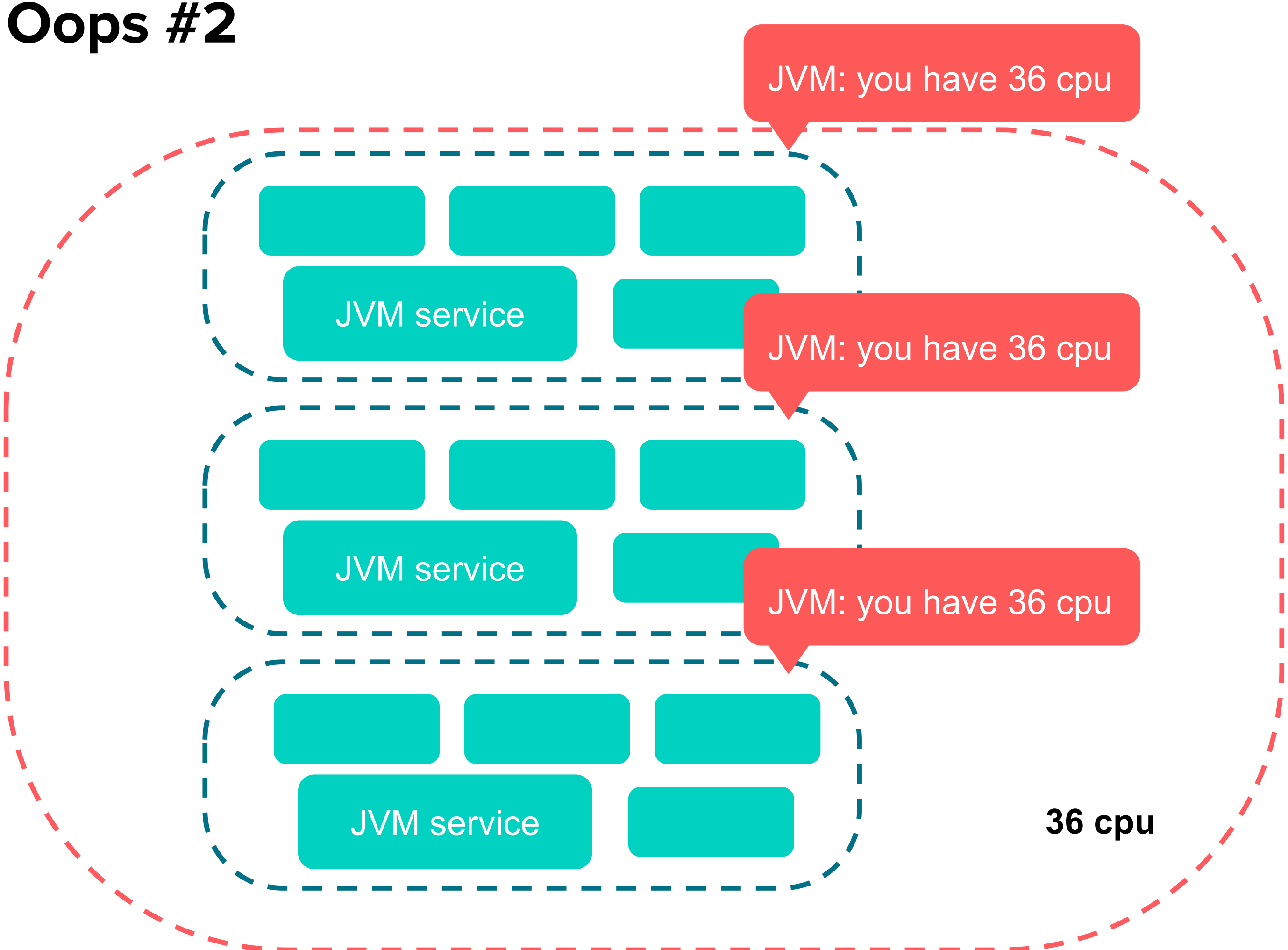
Oops #2

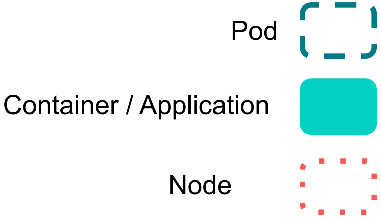




@jiancheung & Stephen Chan

Oops #2





https://kccncna19.sched.com/event/UaVY/10-weird-ways-to-blow-up-your-kubernetes-melanie-cebula-bruce-sherrod-airbnb

Write Once, Run Anywhere

https://bugs.openjdk.java.net/browse/JDK-8146115

Older versions of Java were not “container aware”.

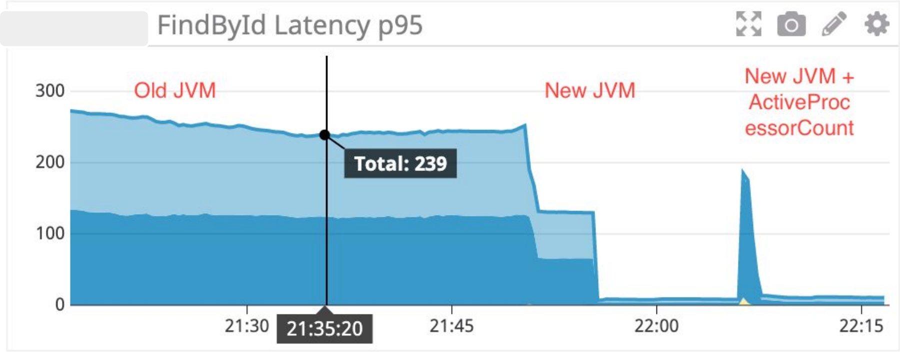
Java tunes itself based on how much resources (like CPU cores) it *thinks* the system has.

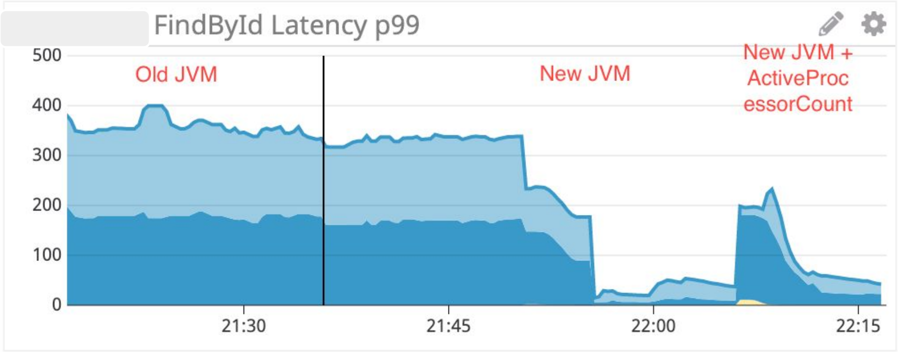
This affects how it tunes things like threadpools, etc

Fixed in Java 8u191+

Lots of posts on this if you google for it

Write Once, Run Anywhere





Also tried playing around with `-XX:ActiveProcessorCount` but it didn’t have much of an effect

Write Once, Run Anywhere

Did Kubernetes make my p95s worse?

Write Once, Run Anywhere

Did Kubernetes make my p95s worse?

YES

Write Once, Run Anywhere

Did Kubernetes make my p95s worse?

YES

… because container’s promise of “Build Once, Run Anywhere” isn’t 100% accurate.

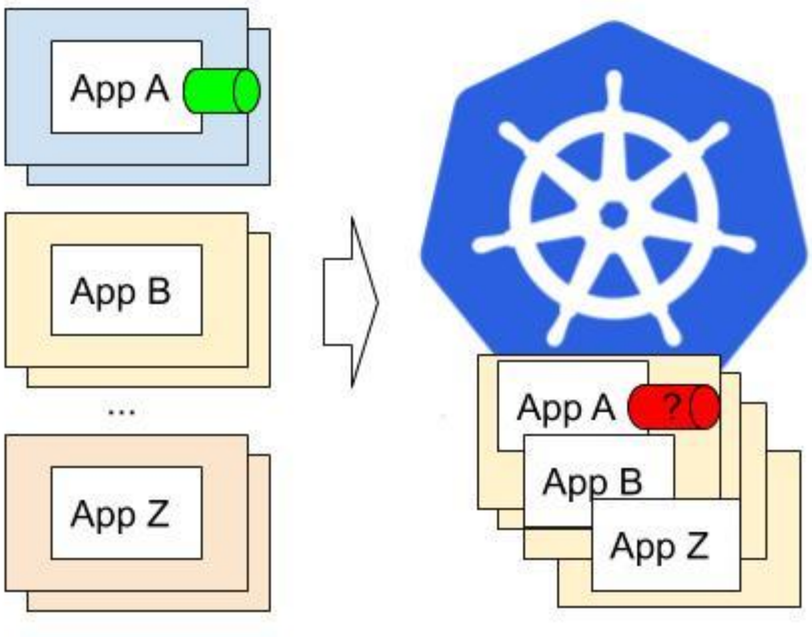
Languages and apps can have deeper dependencies on the underlying systems that they run on.

Write Once, Run Anywhere

Lesson:

Languages and apps can have deep dependencies on the underlying systems that they run on.

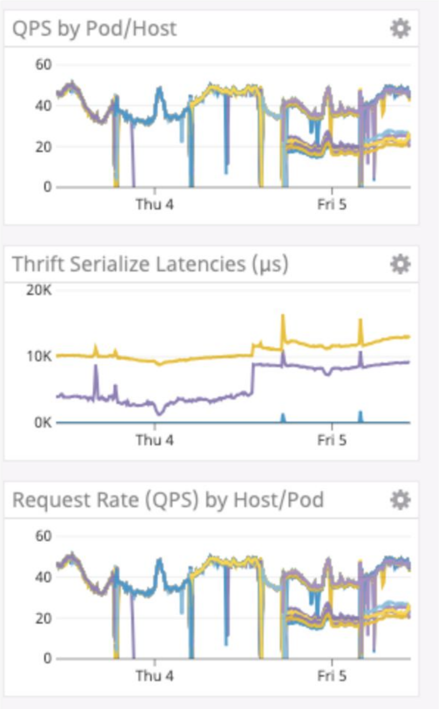
- Upgrade your systems to be “container-aware”
- Having a baseline can be very enlightening



Traffic Imbalance

Traffic Imbalance

For context, ☸☸☸☸ has many pods. We noticed that sometimes on deploys, QPS isn't evenly distributed across the many pods.

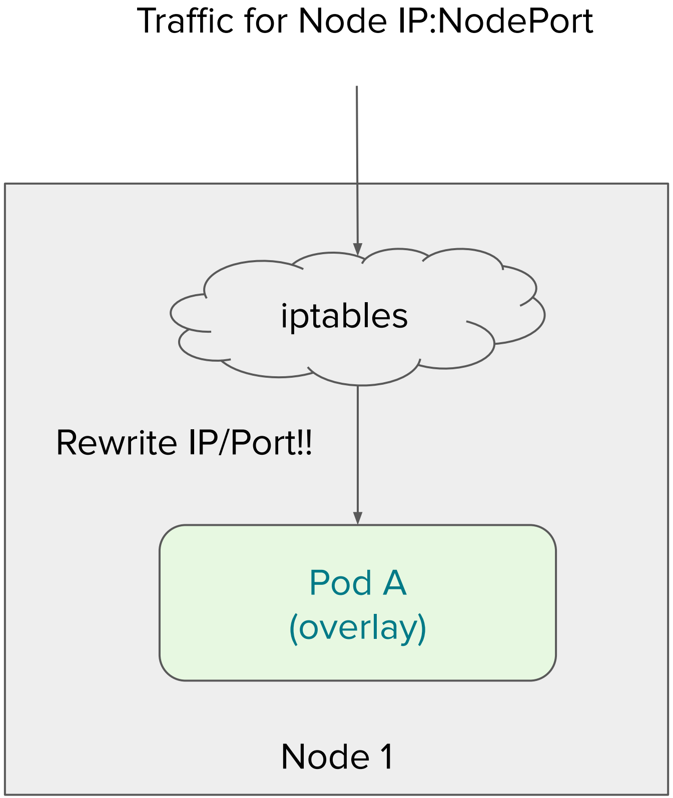


QPS by Pod/Host

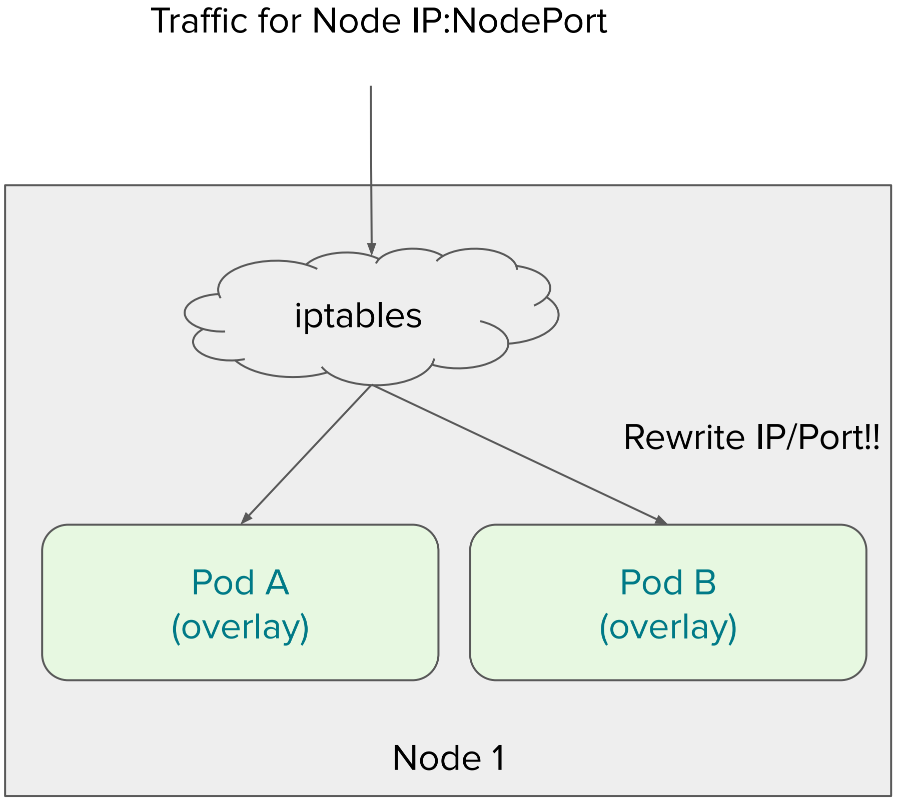
Thrift Serialize Latencies (µs)

Request Rate (QPS) by Host/Pod

Traffic Imbalance



Traffic Imbalance



But which one?
Random

Traffic Imbalance

Did Kubernetes make my p95s worse?

Traffic Imbalance

Did Kubernetes make my p95s worse?

MAYBE

Traffic imbalance causes variable load/latency

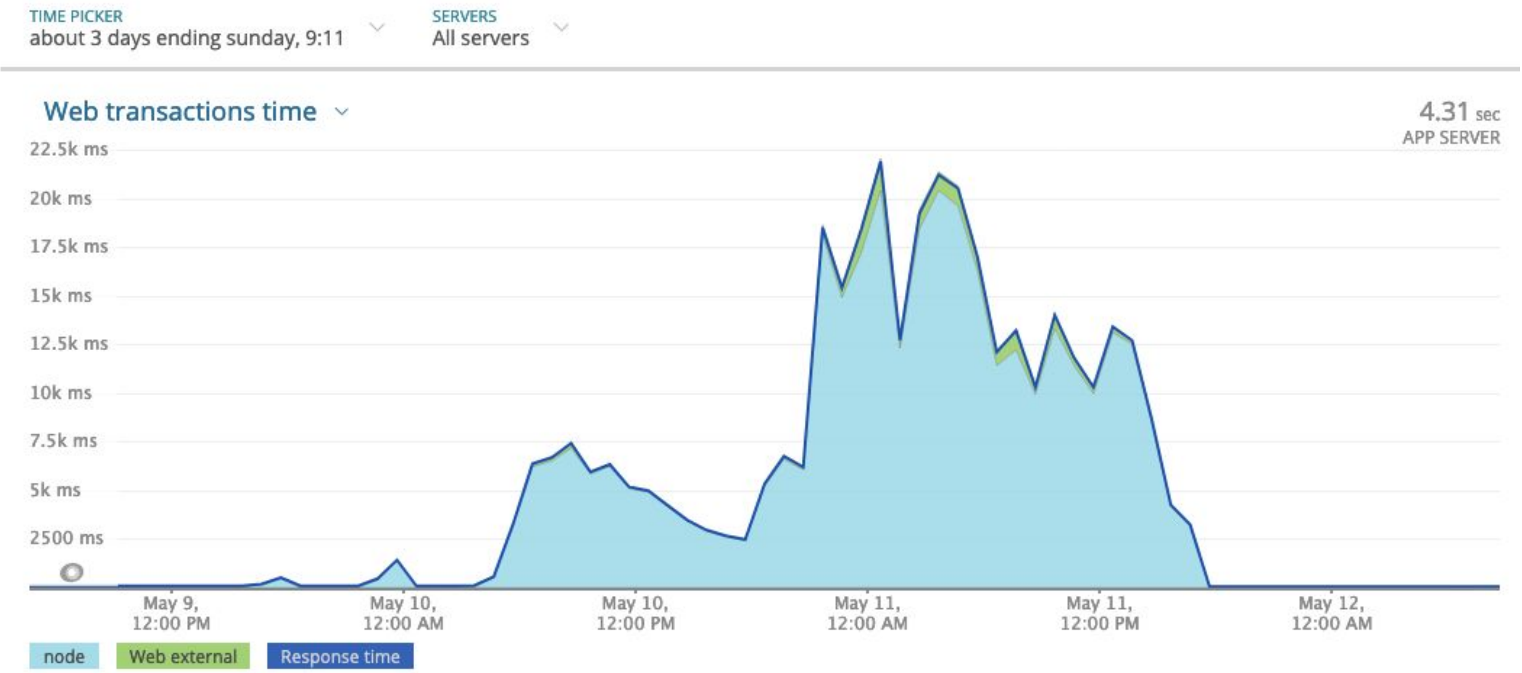
Traffic Imbalance

Lessons:

- Adding an overlay network provides flexibility (less IP capacity planning), but adds complications
- iptables load balancing is not ideal. Consider bypassing by:
    - Using Envoy for balancing between pod IPs
    - Using cloud-provider native IPs to avoid the overlay

Kube DNS slowness

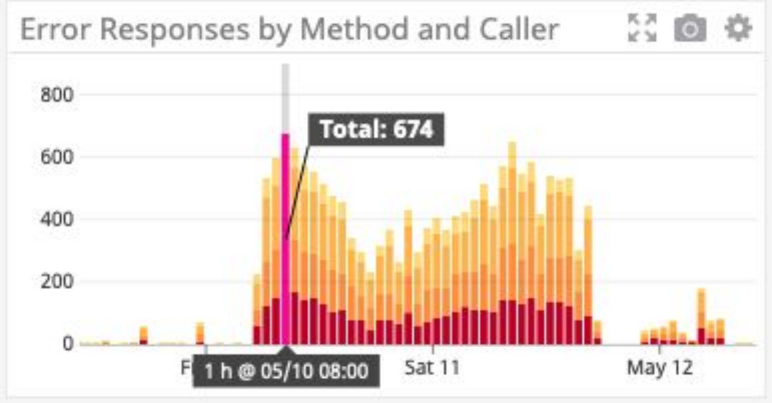
Kube DNS slowness

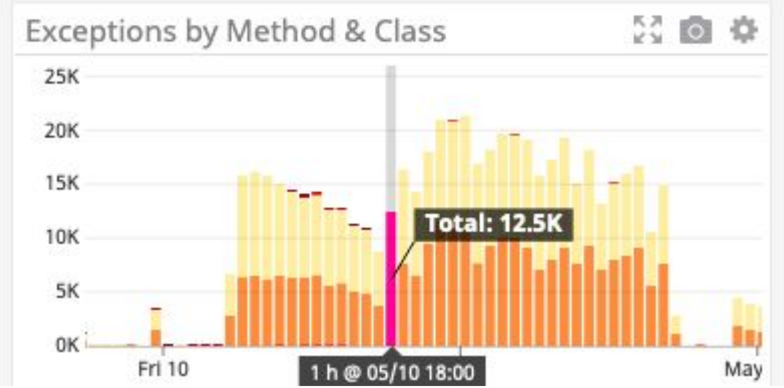


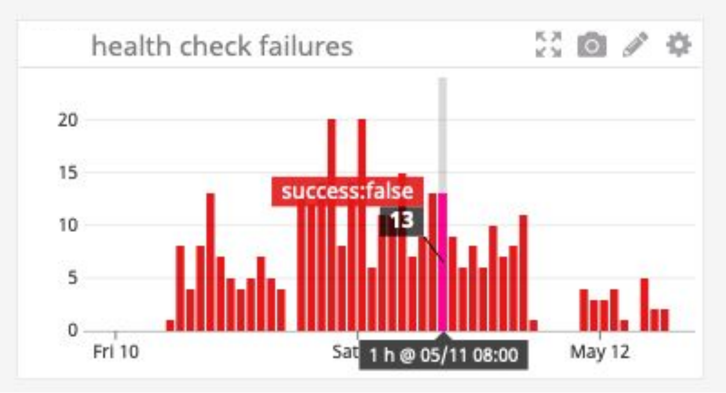
TIME PICKER
about 3 days ending sunday, 9:11

SERVERS
All servers

Kube DNS slowness

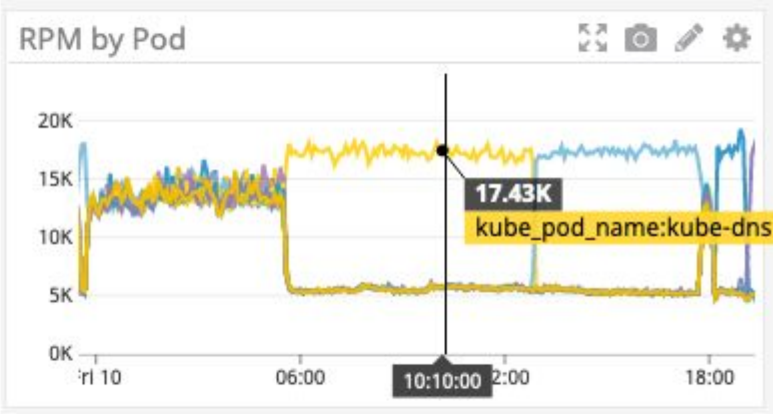






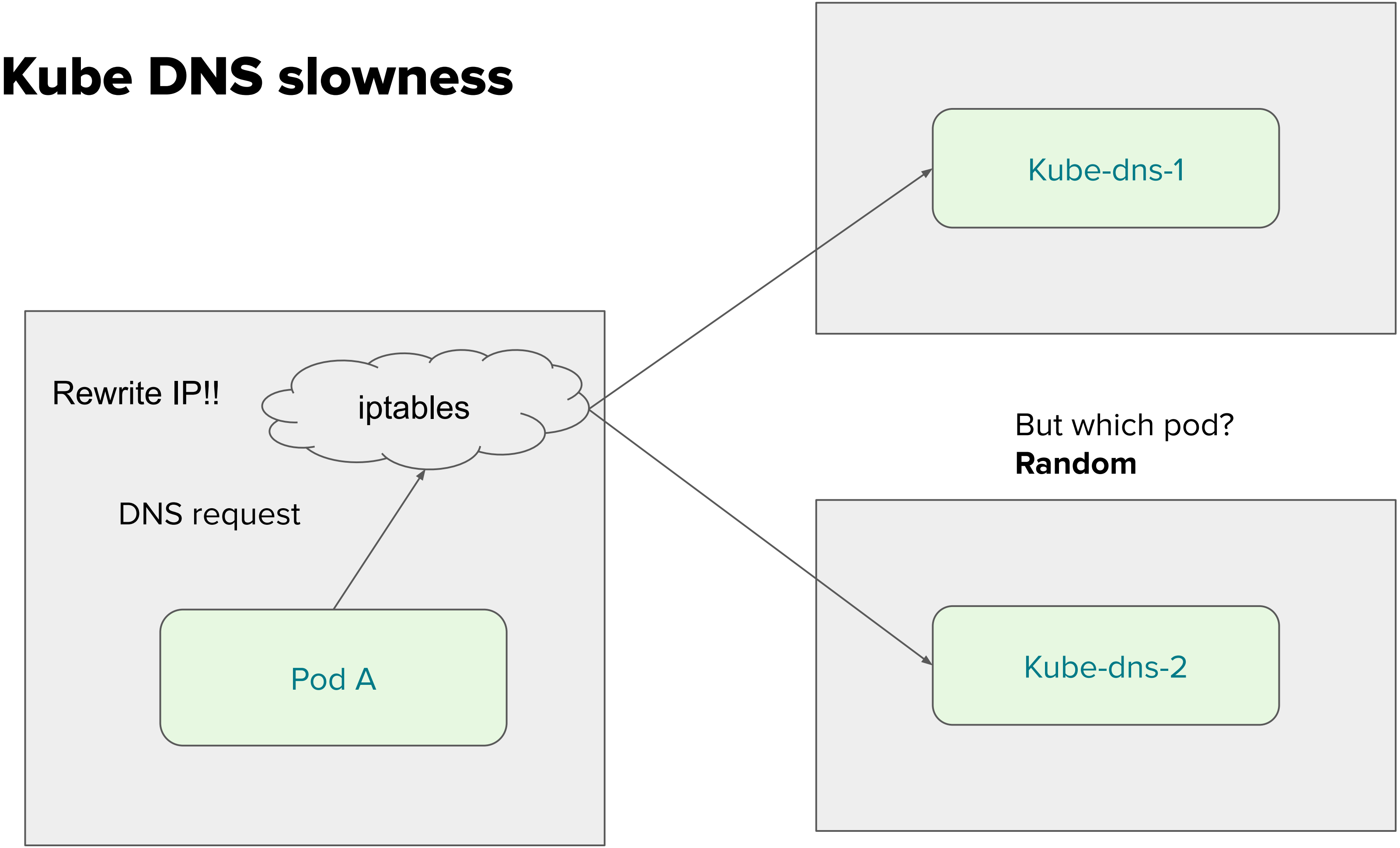
@jiancheung & Stephen Chan

Kube DNS slowness



Traffic imbalance strikes again!

Kube DNS slowness



Rewrite IP!!

iptables

DNS request

Pod A

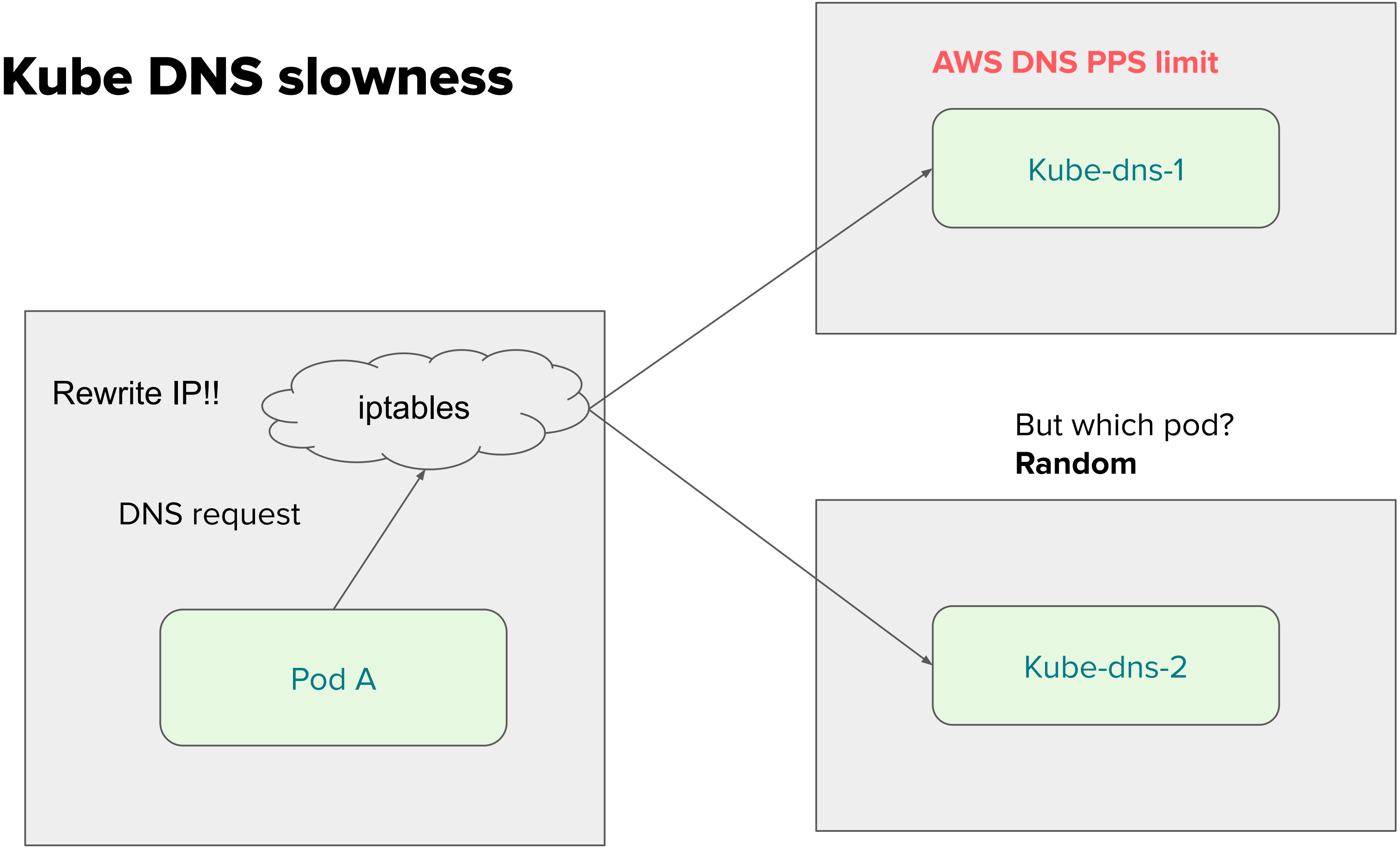
Kube-dns-1

But which pod?
Random

Kube-dns-2

@jiancheung & Stephen Chan

Kube DNS slowness


AWS DNS PPS limit

@jiancheung & Stephen Chan

Kube DNS slowness

Did Kubernetes make my p95s worse?

Kube DNS slowness

Did Kubernetes make my p95s worse?

YES

Kube DNS slowness

Lessons:

- By default, kube-dns is discovered through ClusterIP (more potential iptables imbalance!)
- If your pods don’t need Kubernetes DNS resolution, set pod dnsPolicy to `Default` (or `None` if customization needed)

Recap

Recap

Case	Did K8s P95s worse?	Lessons
Latencies *Improved*?		
Noisy Neighbors		
Noisy Neighbors, made worse by K8s		
Write Once, Run Anywhere		
Traffic Imbalance		
Kube DNS slowness		

Recap

Case	Did K8s P95s worse?	Lessons
Latencies *Improved*?	No (but take the credit if perf improved 😇)	Underlying systems affect performance like hardware, host OS, etc.
Noisy Neighbors		
Noisy Neighbors, made worse by K8s		
Write Once, Run Anywhere		
Traffic Imbalance		
Kube DNS slowness		

Recap

Case	Did K8s P95s worse?	Lessons
Latencies *Improved*?	No (but take the credit if perf improved 😇)	Underlying systems affect performance like hardware, host OS, etc.
Noisy Neighbors	Yes	Set limits. Be wary of how CPU is counted.
Noisy Neighbors, made worse by K8s		
Write Once, Run Anywhere		
Traffic Imbalance		
Kube DNS slowness		

@jiancheung & Stephen Chan

Recap

Case	Did K8s P95s worse?	Lessons
Latencies *Improved*?	No (but take the credit if perf improved 😇)	Underlying systems affect performance like hardware, host OS, etc.
Noisy Neighbors	Yes	Set limits. Be wary of how CPU is counted.
Noisy Neighbors, made worse by K8s	Yes	Tune your priorities and predicates!
Write Once, Run Anywhere		
Traffic Imbalance		
Kube DNS slowness		

Recap

Case	Did K8s P95s worse?	Lessons
Latencies *Improved*?	No (but take the credit if perf improved 😇)	Underlying systems affect performance like hardware, host OS, etc.
Noisy Neighbors	Yes	Set limits. Be wary of how CPU is counted.
Noisy Neighbors, made worse by K8s	Yes	Tune your priorities and predicates!
Write Once, Run Anywhere	Yes ish (move to containers did)	Upgrade apps/languages to be “container-aware”.
Traffic Imbalance		
Kube DNS slowness		

Recap

Case	Did K8s P95s worse?	Lessons
Latencies *Improved*?	No (but take the credit if perf improved 😇)	Underlying systems affect performance like hardware, host OS, etc.
Noisy Neighbors	Yes	Set limits. Be wary of how CPU is counted.
Noisy Neighbors, made worse by K8s	Yes	Tune your priorities and predicates!
Write Once, Run Anywhere	Yes ish (move to containers did)	Upgrade apps/languages to be “container-aware”.
Traffic Imbalance	Maybe	Be wary of iptables load-balancing
Kube DNS slowness		

Recap

Case	Did K8s P95s worse?	Lessons
Latencies *Improved*?	No (but take the credit if perf improved 😇)	Underlying systems affect performance like hardware, host OS, etc.
Noisy Neighbors	Yes	Set limits. Be wary of how CPU is counted.
Noisy Neighbors, made worse by K8s	Yes	Tune your priorities and predicates!
Write Once, Run Anywhere	Yes ish (move to containers did)	Upgrade apps/languages to be “container-aware”.
Traffic Imbalance	Maybe	Be wary of iptables load-balancing
Kube DNS slowness	Yes	Check your dnsPolicy early and often

Other Takeaways

- Performance includes tuning at all layers of the stack (host, cluster, container, application, language)
- Set expectations that small performance differences can happen
- Having a baseline can be useful to even be aware of performance gains

Thanks!

- learn more @ medium.com/airbnb-engineering
- jobs @ airbnb.com/careers