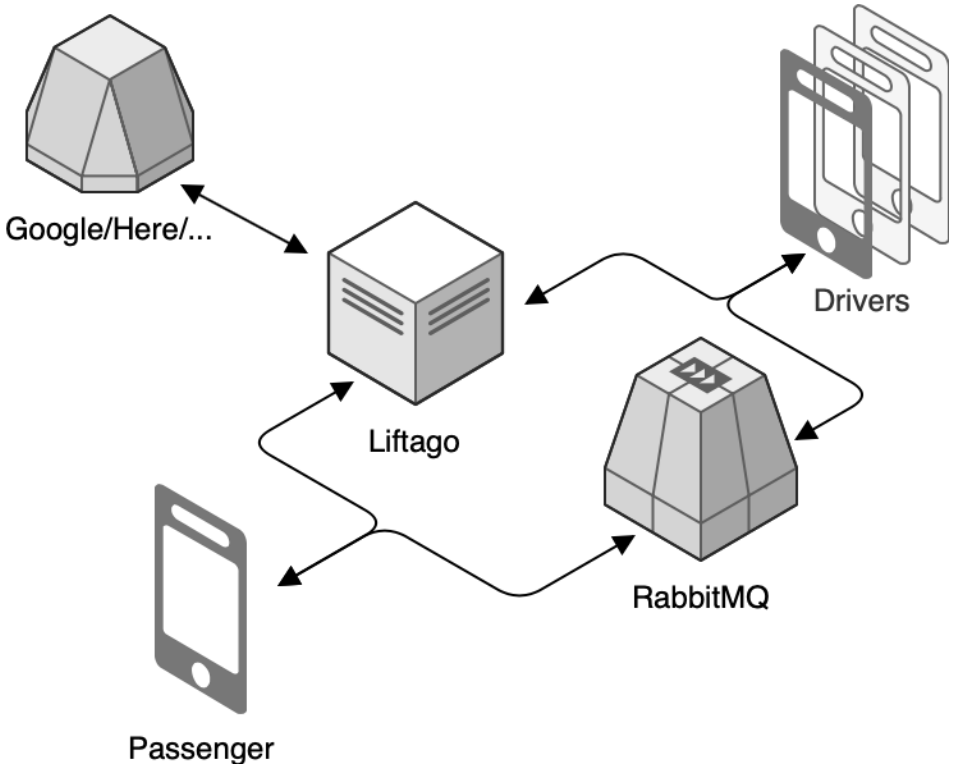
Liftago

Operations at Liftago

kubernetes

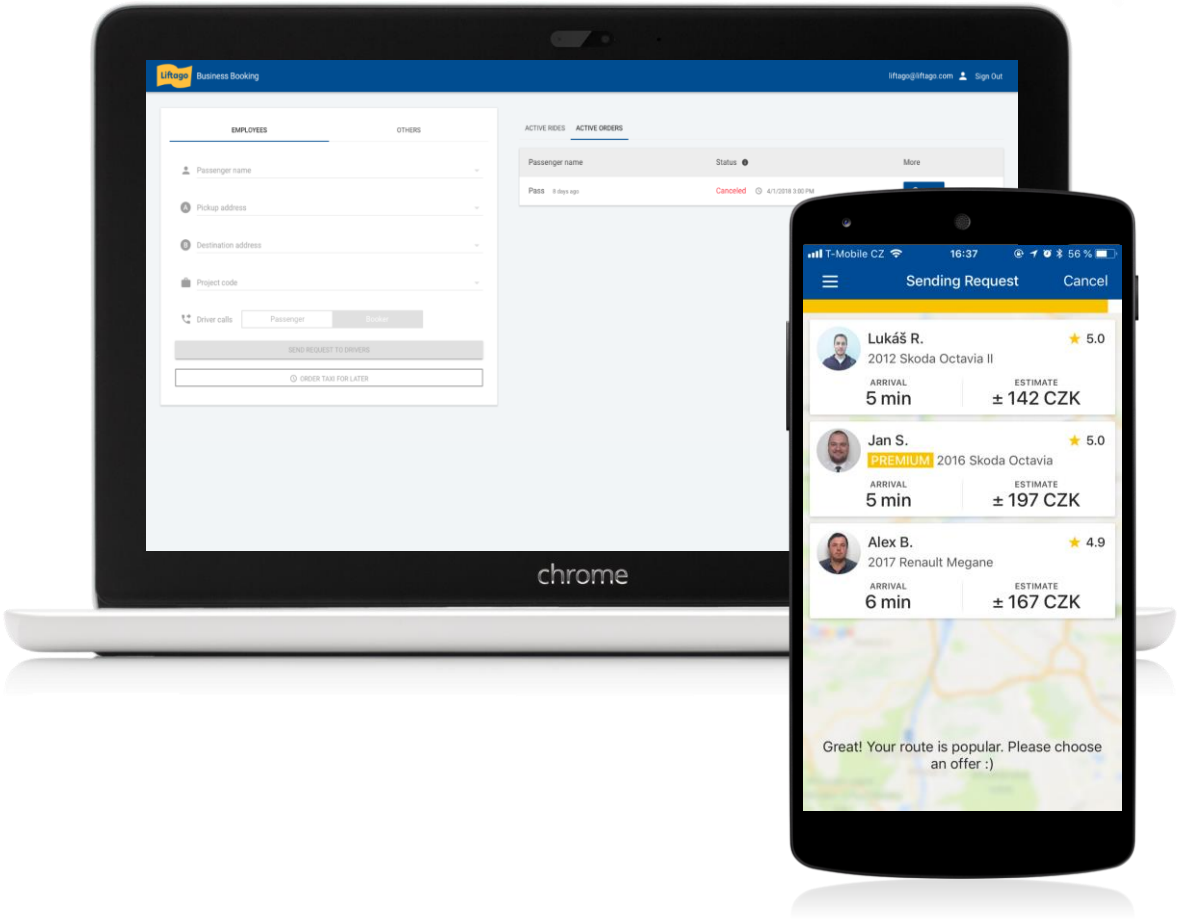
Radek Los


Google/Here/...
Drivers
Liftago
RabbitMQ
Passenger

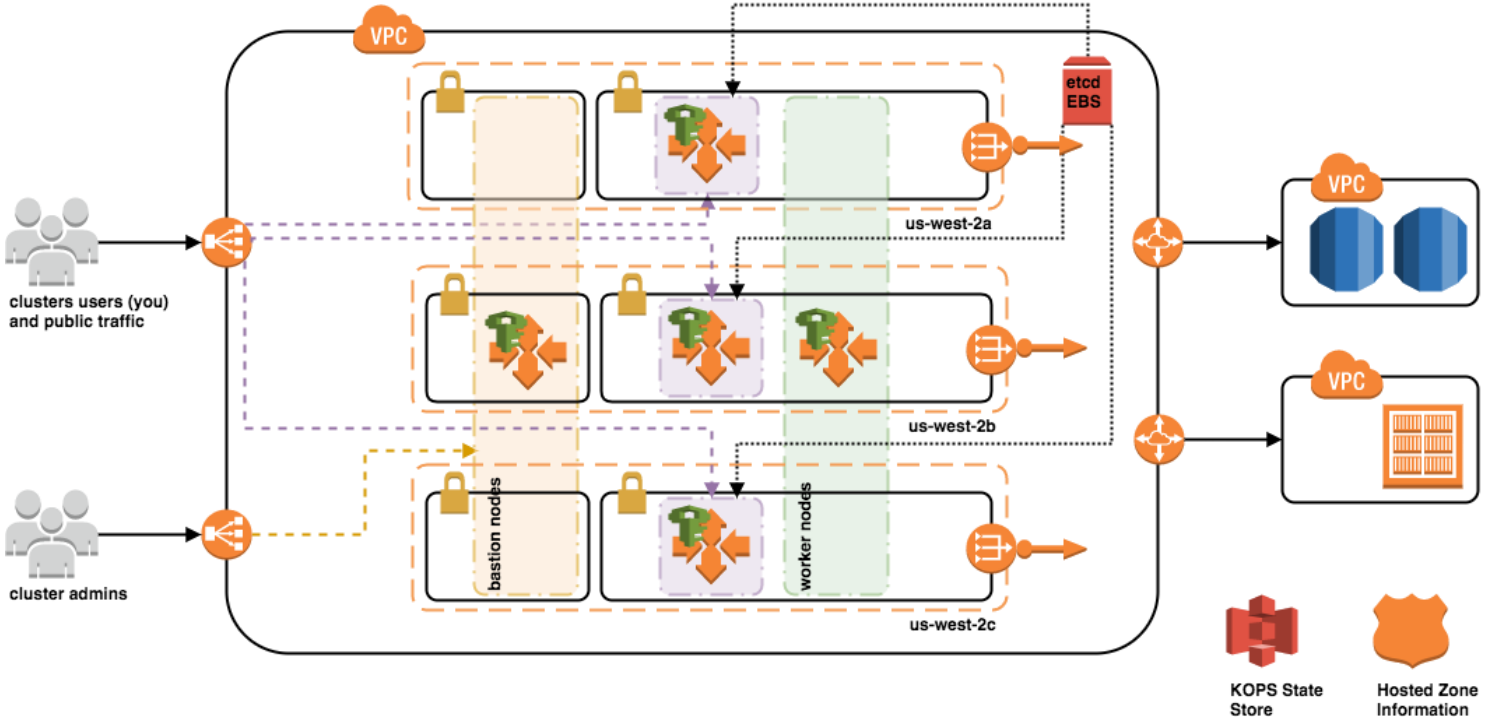
Our Engineering

- iOS/Android/Web/APIs
- Web for booking rides
- Web for managing business
- Internal Admin
- Fleet management
- Accounting
- Taxi management

- AWS
- K8s (docker/java/js)
- Multi-AZ/HA/self healing cluster
- CI/CD to dev/test
- Daily deployments to PROD



Production - Kubernetes



https://www.joshdurbin.net/posts/2018-04-kubernetes-aws-kops/

Devs	Ops
4	0

Running Kubernetes cluster by Java devs?

KOPS

What is Kops?

- Fully automated installation
- Uses DNS to identify clusters
- Self-healing: everything runs in Auto-Scaling Groups
- Limited OS support (Debian preferred, Ubuntu 16.04 supported, early support for CentOS & RHEL)
- High-Availability support (Drain and Validate Rolling Update)
- Can directly provision, or generate terraform manifests

Pre-requirements

- Install kops + awscli
- Create a route53 domain for your cluster
- Create an S3 bucket to store your clusters state
    - Bucket name is stored in environment variable (KOPS_STATE_STORE)

Create cluster

```
kops --v 3 create cluster \
   --cloud aws \
   --name $CLUSTER_NAME \
   --zones eu-central-1a,eu-central-1b,eu-central-1c \
   --node-size t2.medium \
   --node-count 5 \
   --master-zones eu-central-1a \
   --master-size m3.medium \
   --network-cidr 172.10.0.0/16 \
   --topology private \
   --networking kopeio-vxlan \
   --bastion \
   --ssh-public-key ssh/devtest.pub
```

Create cluster

```
kops update cluster $CLUSTER_NAME --yes
```

Demo


HilariousGifs.com
It's working, It's working!

Outages

- I don’t want to be woken up
- I want to to go sleep as soon as possible
- I don’t want to be woken up again
  - Post mortem
  - Fix it this or next sprint

2019/01/30

K8S master restart

2018/12/15

Stuck reading from Taxi Position

2018/07/12

2018/07/12 - Failed mongo update

2018/01/26

2018/01/26 Outage - APNS Dev certificate

2017/10/20

2017/10/21 Outage analysis

30. 01. 2018 - K8s master restart

Radek Los 12:38 PM

@here dela nekdo neco na produkcnim clusteru? vypada ze se restartoval k8s master..

Tibor Mlynárik 12:38 PM

ja nie

vyzera to na vypadok

Peter Nikodem 12:41 PM

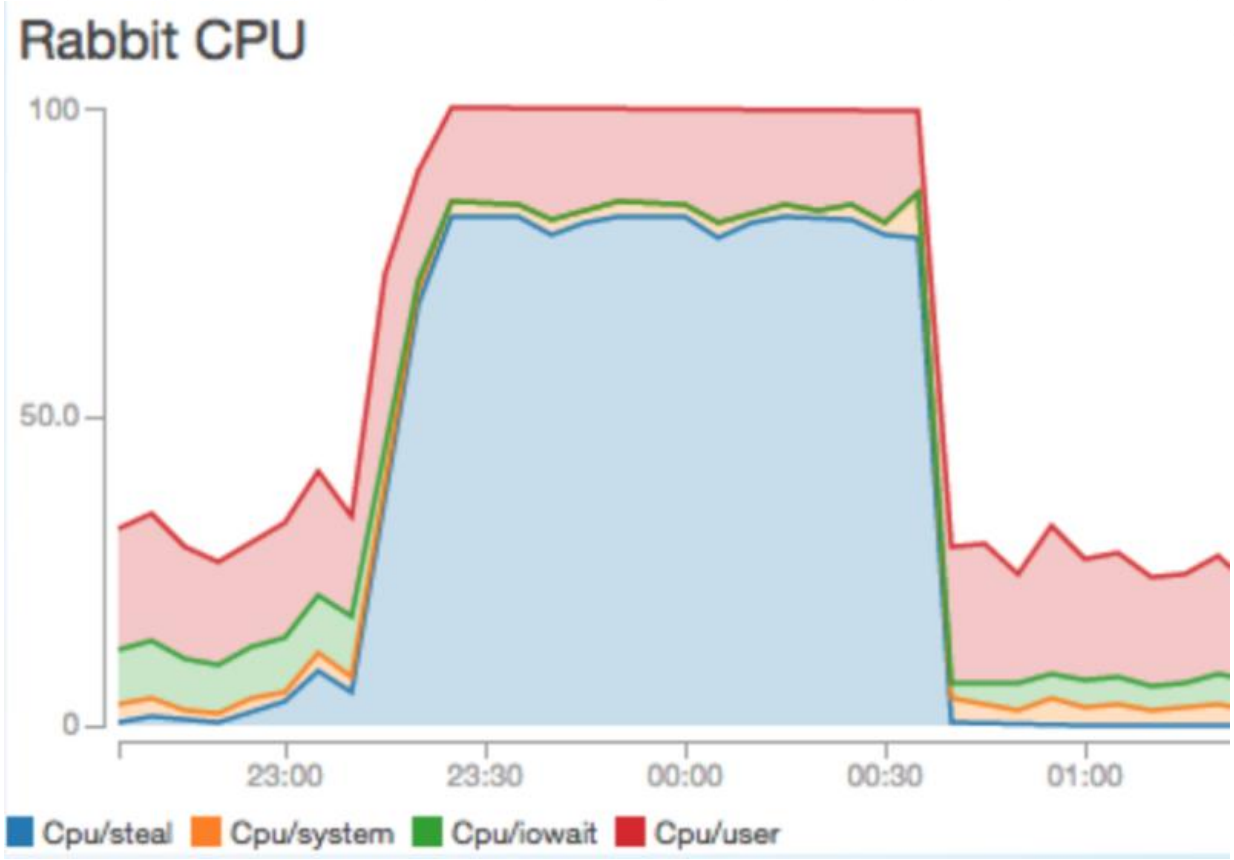
Volalo mi PD

If the Status of the Ready condition remains **Unknown** or **False** for longer than the **pod-eviction-timeout**, an argument is passed to the kube-controller-manager and all the Pods on the node are scheduled for deletion by the Node Controller. The default eviction timeout duration is **five minutes**. In some cases when the node is unreachable, the apiserver is unable to communicate with the kubelet on the node. The decision to delete the pods cannot be communicated to the kubelet until communication with the apiserver is re-established. In the meantime, the pods that are scheduled for deletion may continue to run on the partitioned node.

https://kubernetes.io/docs/concepts/architecture/nodes/

9. 12. 2016 - Rabbit MQ

- After 5 days of successful migration to CloudAMQP
  - 00:10 - TaxiPosition queue is growing, CPU usage started growing at similar time
  - 00:25 - Increasing number of connections on MQTT
  - 00:31 - PagerDuty alert triggered by CloudAMQP CPU usage
  - 00:59 - PagerDuty alert triggered by number of connections
  - 01:30 - RabbitMQ upgrade to higher plan
  - 01:40 - Everything is operating normally


Rabbit CPU
100
50.0
0
23:00
23:30
00:00
00:30
01:00
Cpu/steal
Cpu/system
Cpu/iowait
Cpu/user

6. AWS BIG BUNNY USERS WITH HIGH STEAL TIME

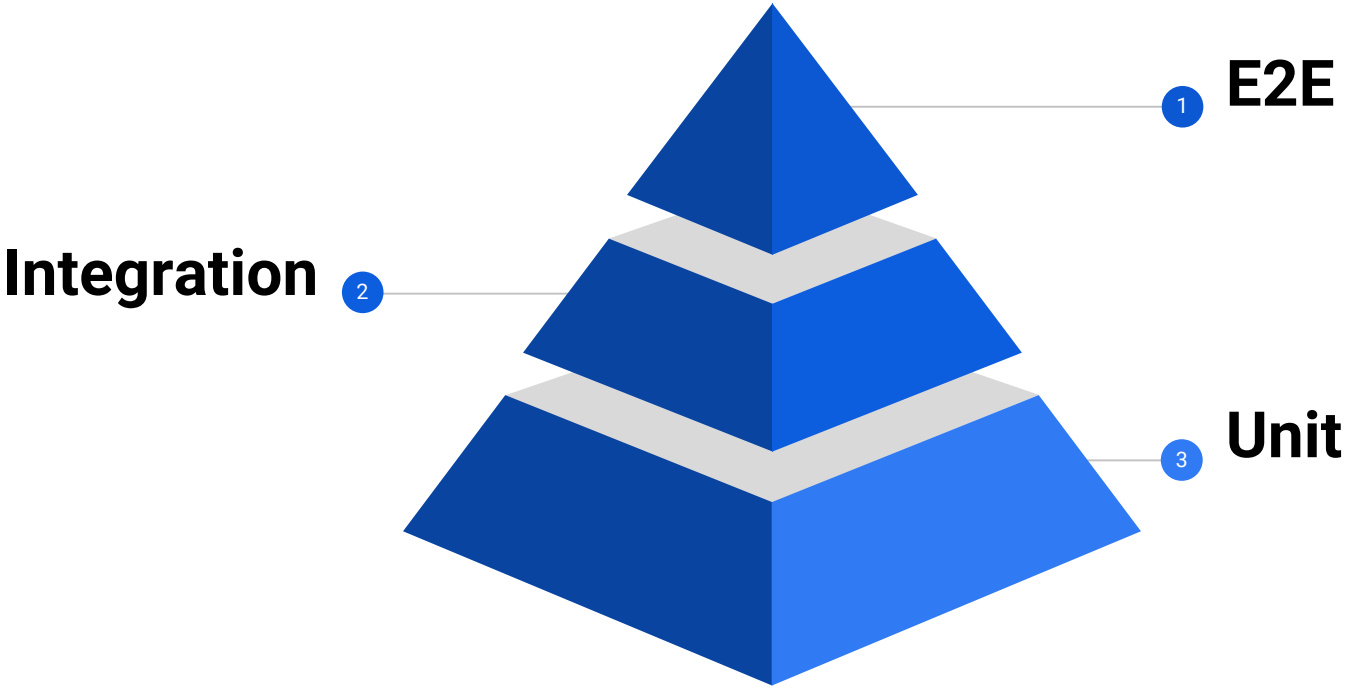
If you are on plan Big Bunny on AWS you are on T2-instances. If you, for example, have too high message throughput during a too long time you will run out of CPU credits. When that happens, **steal time** is added and your instance starts throttle (your instance will be running slower than it should). If you notice high steal time you need to upgrade to a larger instance or identify the reason for your CPU usage.

https://www.cloudamqp.com/blog/2016-01-25-identify-and-protect-against-high-cpu-and-memory-usage.html

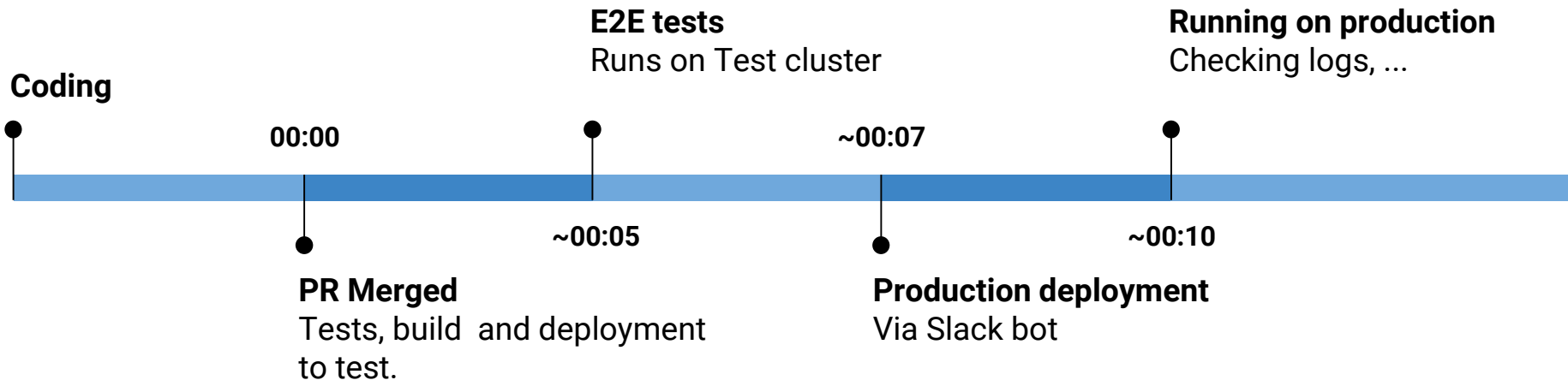
Lesson learnt

- Know your hosted environment
- We met limits of T2 instances
- We read some docs

Tests pyramid



Pipeline


Coding
00:00
PR Merged
Tests, build and deployment to test.
E2E tests
Runs on Test cluster
~00:05
~00:07
Production deployment
Via Slack bot
Running on production
Checking logs, ...
~00:10

Demo

Two years with Kops

- Several upgrades on devtest cluster on runtime without “downtime”
- Patch upgrade on production
- AWS credits (CPU utilization) run out on PROD
- Network splits, zones issues, EC2 death
- Easy management when your cluster grows
- No further configurations
- Keeps AWS clean

?

@radeklos

@liftago_eng

https://medium.com/liftago-engineering

Thank you for coming!

bit.ly/cnp_feedback