Amazon Elastic Container Service for Kubernetes (EKS), AWS Fargate and Beyond.

Mency Woo

TriNimbus
Cloud Management Solutions

Another talk about containers?

Show apps

VIDEOS SMART CITIES WINDOWS 10 CLOUD INNOVATION SECURITY TECH PRO MORE NEWSLETTERS ALL WRITERS

JUST IN WHAT IS V2X COMMUNICATION? CREATING CONNECTIVITY FOR THE AUTONOMOUS CAR ERA

What is Docker and why is it so darn popular?

Docker is hotter than hot because it makes it possible to get far more apps running on the same old servers and it also makes it very easy to package and ship programs. Here's what you need to know about it.

By Steven J. Vaughan-Nichols for Linux and Open Source | May 8, 2017 -- 12:32 GMT (05:32 PDT) | Topic: Cloud

OCT 16, 2016 @ 06:10 PM 8,570

The Little B

The Point Of Docker Is More Than Containers

Justin Warren, CONTRIBUTOR

Justin is Chief Analyst at PivotNine consultants and advisors.

FULL BIO

Opinions expressed by Forbes Contributors are their own.

I was a Docker skeptic. The idea of containers was not particularly new (BSD Jails, Solaris zones, etc.) and it didn't seem to be *enough* to sustain a company or a new way of working.

The LEGO Docker whale at Docker[+]

I've changed my mind.

TechRepublic. SEARCH

Digital Transformation Cloud Big Data AI IoT More Newsletters Forums Resource Library Tech

CLOUD

Netflix on containers: Five ways they'll boost our business, from lower costs to higher productivity

Learn why your business might want to emulate Netflix and rearchitect back-end systems around Docker containers.

By Nick Heath | September 12, 2016, 5:12 AM PST

CIO JOURNAL.

Royal Caribbean CIO Deploys Containers to Modernize Decades-Old Reservation System

The goal is faster, more personalized service for online and mobile cruise-goers

By Kim S. Nash

Dec 14, 2016 6:12 pm ET

6 COMMENTS

Recommended Videos

Q&A

What are containers and why do you need them?

Containers are a solution to the problem of how to get software to run reliably when moved from one computing environment to another. Here's what you need to know about this popular technology.

By Paul Rubens

CIO | JUN 27, 2017 3:00 AM PT

Disclaimer

- AWS literature and diagrams are referenced
- Third Party resources are annotated where appropriate
- My opinions are but my own
- Errors are my sole responsibility
- **Loads of content, discussions and questions to the very end**
- **Each subtopic can merit its own deep dive session, feedback on further discussions appreciated**

Challenge

- How do I explain to my CEO why he/she should care about container?
- How do I know which container tech to choose, when the tech is moving so quickly?
- How do I “just get it to work” ?

Bottomline

- Container is popular; and it is getting even more popular
- There are many benefits, but it is not free
- Running container is easy. Running containers in scale is NOT.
- Container are good at solving some problems, but it is NOT panacea to all problems

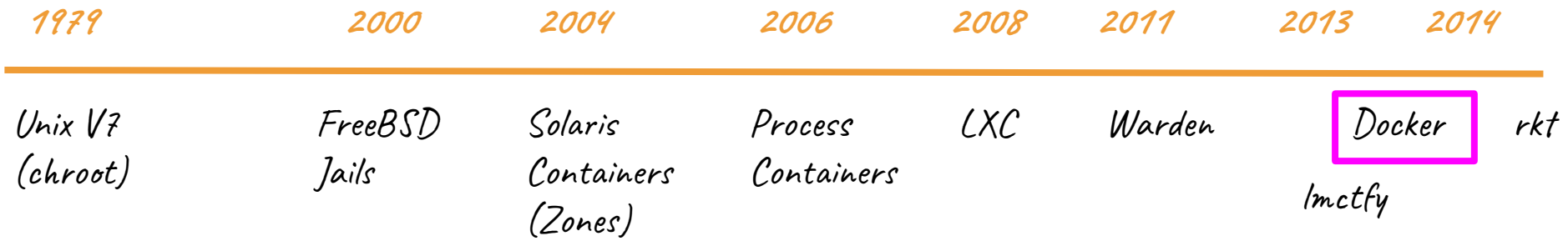
Agenda

- **What** and **Why**: Overview on Containers
- **How**: Overview and Demo of AWS Container Services
  - EKS (still in preview in us-west-2)
  - Fargate (GA in us-east-1)
  - ECS (available in most regions)
- **When** should I use which to run container workload on AWS?

Containers / Dockers

Containerization

- is hot topic, but not new
- has been a concept since …
  before most of us are born …



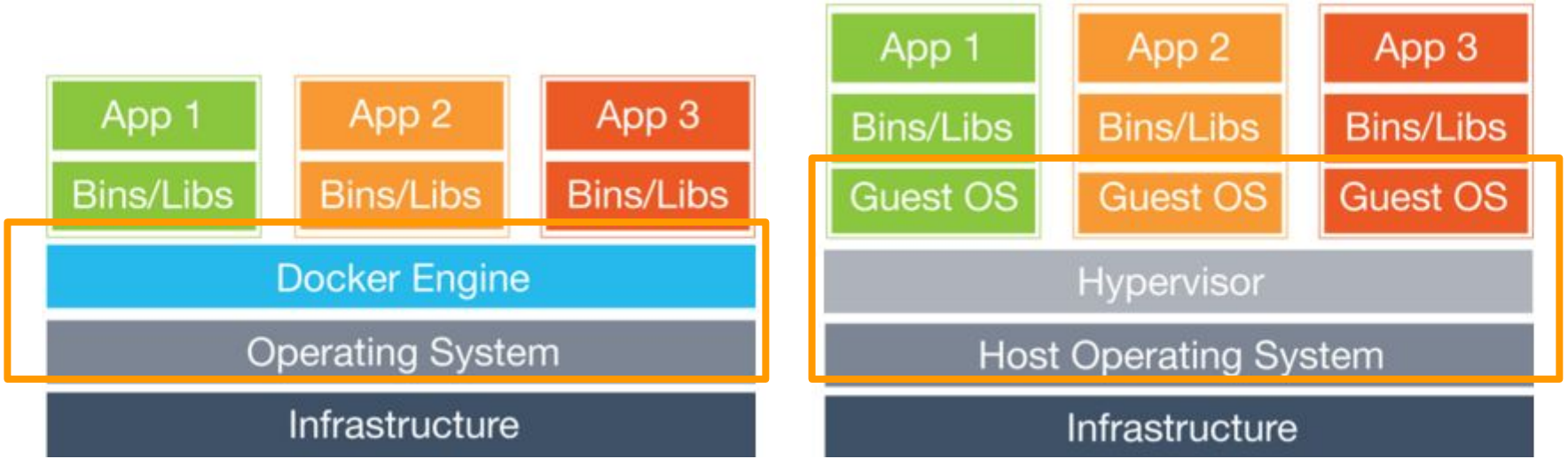
Source: https://blog.aquasec.com/a-brief-history-of-containers-from-1970s-chroot-to-docker-2016

Why Docker/Container?

- Portable
- Simple
- Dependency Management *
- Community Adoption *
- Speed *
- Cost efficiency
- Modularity / Scalability

Containerization vs Virtualization

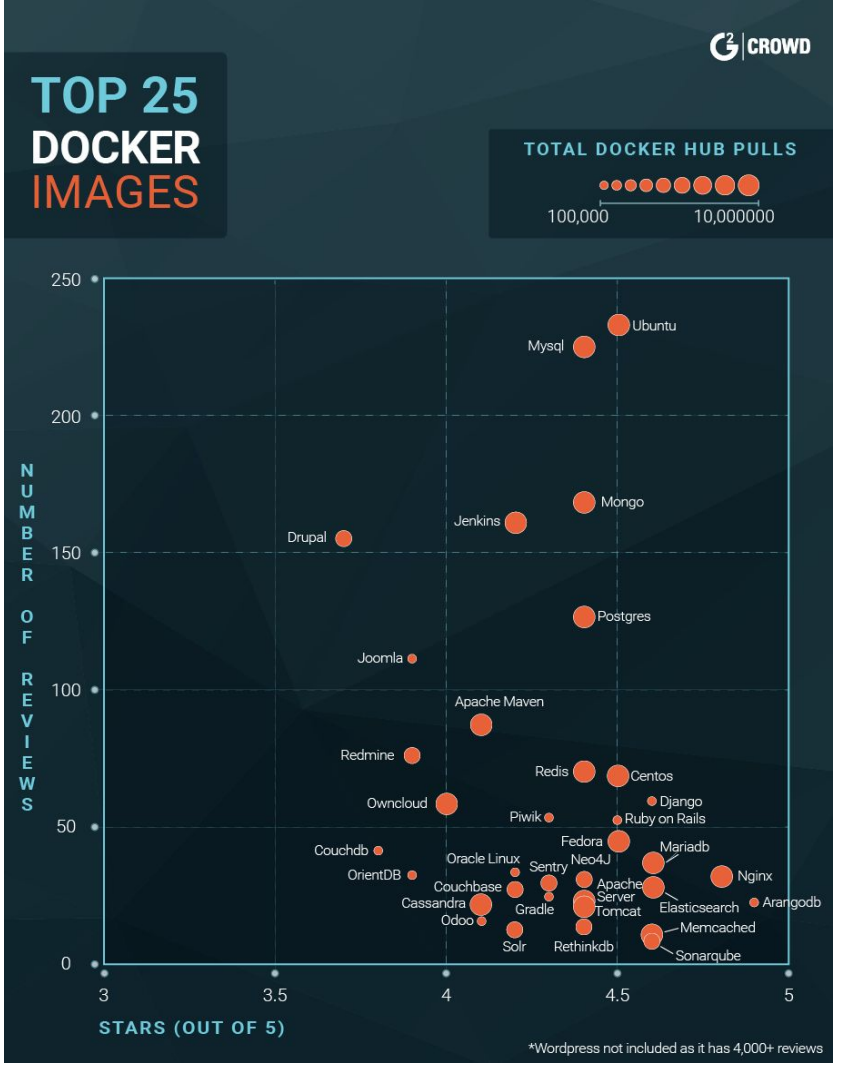
- OS virtualization vs hypervisor virtualization
- Lightweight (vs heavyweight)
- Shared kernel space vs emulated kernels**



Source: https://cdn-images-1.medium.com/max/1600/1*cshk5nzsiaUsBmy2Jq_Guw.png

Community Support

- Docker hub
  https://hub.docker.com/explore/
  - 100,000+ public apps
  - 10,000,000 pulls on popular apps
- Amazon Elastic Container Registry
  https://aws.amazon.com/ecr/
- Google Cloud Registry (gcr.io)

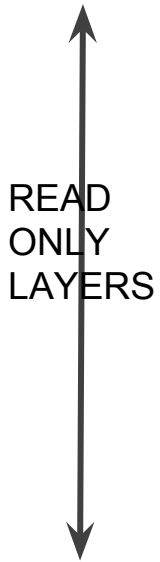


Source:
https://blog.g2crowd.com/blog/containerization/best-apps-images-repositories-docker-hub-2017/

Why is it so fast?

Maximize re-usability/speed by separate layers cleverly!!

```
FROM node
RUN apt-get update
RUN mkdir -p /opt/webapp

WORKDIR /opt/webapp
COPY ./webapp/package.json /opt/webapp/

RUN npm install

COPY ./webapp/bin/  /opt/webapp/bin/
COPY ./webapp/public/stylesheets
/opt/webapp/public/stylesheets

COPY ./webapp/routes/  /opt/webapp/routes/
COPY ./webapp/views/  /opt/webapp/views/
COPY ./webapp/app.js /opt/webapp/

EXPOSE 8080
CMD [ "npm", "start" ]
```



5ad31416f4da : write container layer: docker run sample-image

78b742d8485e : image layer: CMD ["npm", "start"]

…

8150b237ebb6 : image layer: WORKDIR /opt/webapp

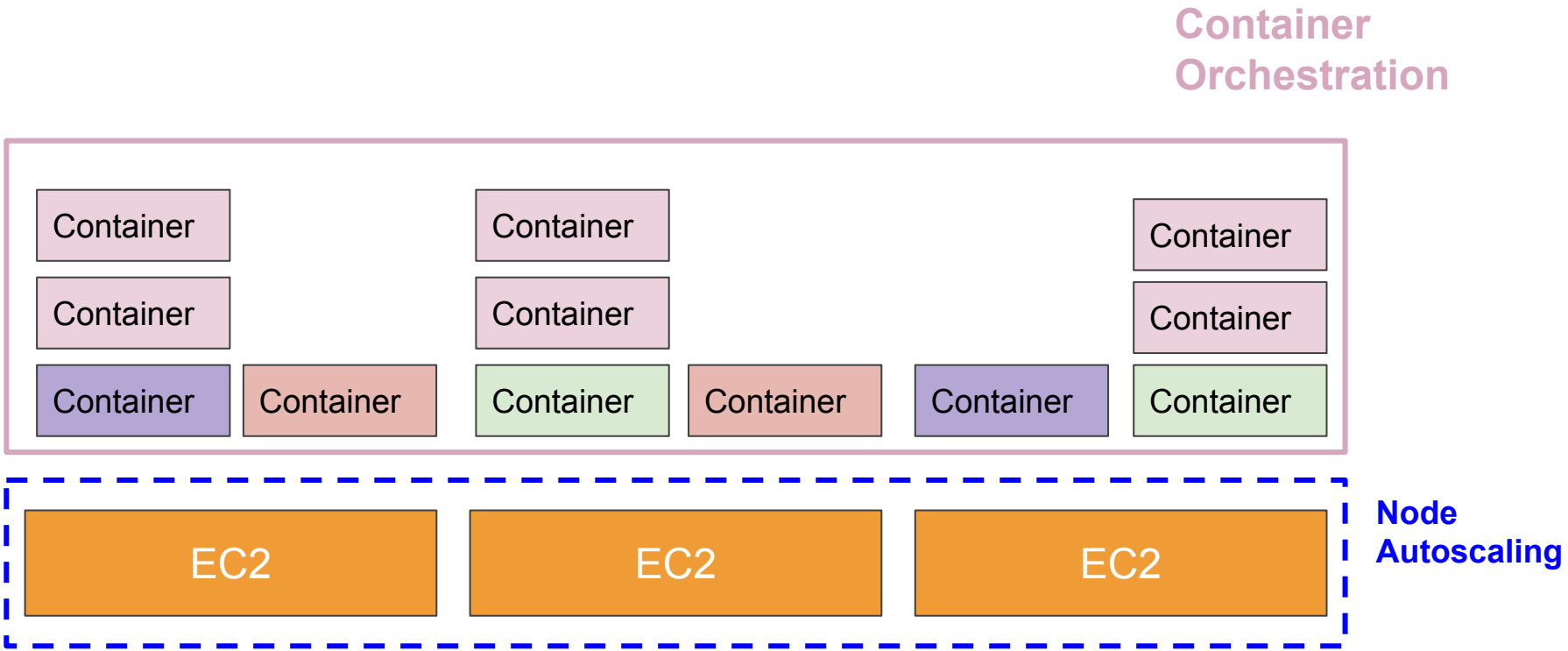
40c87255e8dd : image layer: RUN mkdir -p /opt/webapp

630a7e56fbed : image layer: RUN apt-get update

993f38da6c6c : base image: node

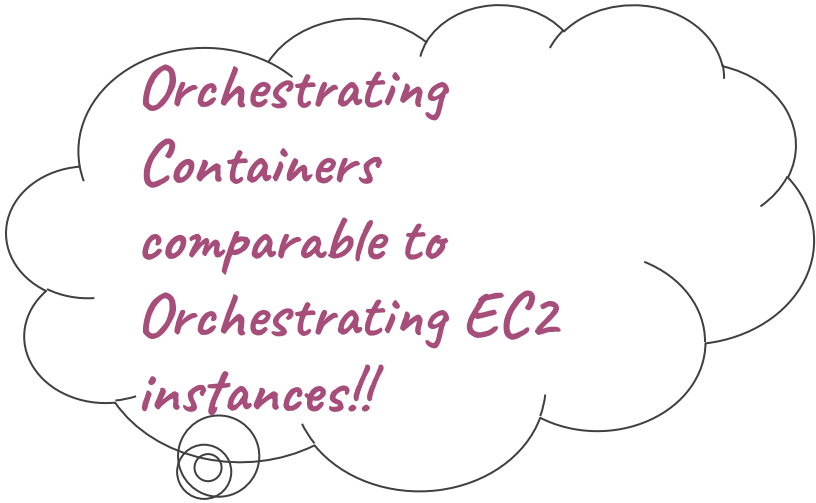
Container is only half the business

"There ain't no such thing as a free lunch"

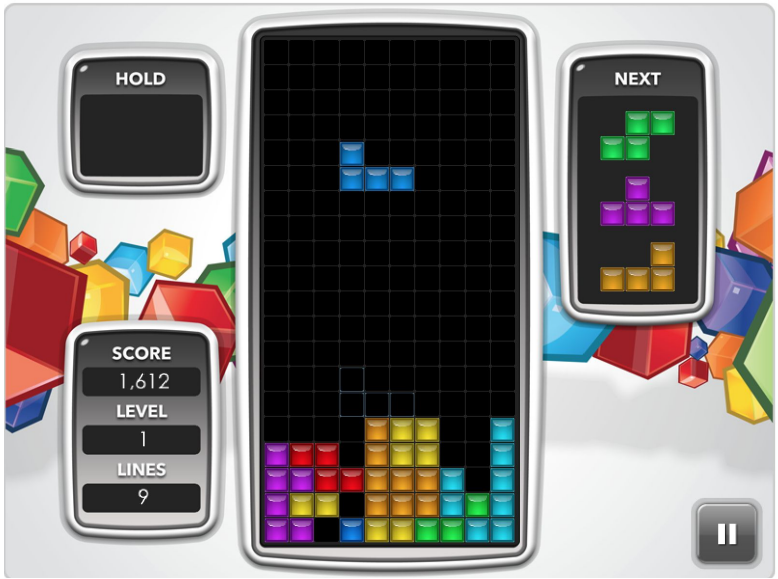


Container Orchestration includes

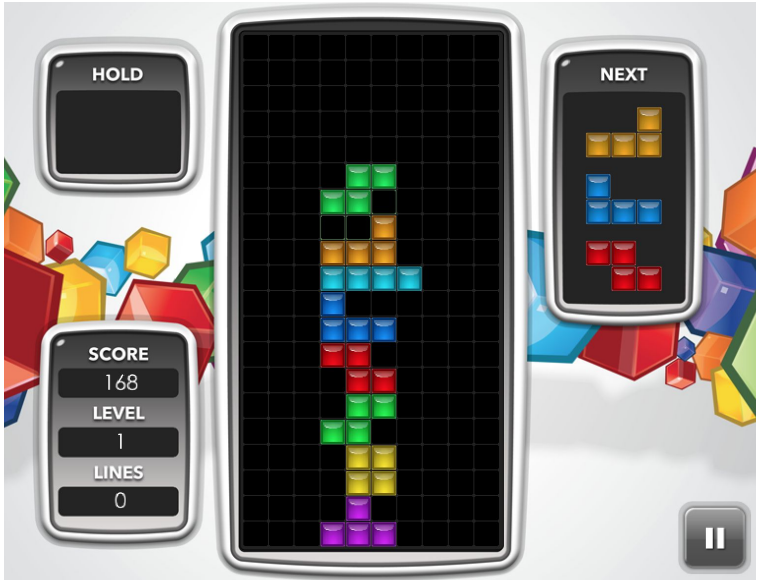
- Provisioning hosts
- Starting/Instantiating containers on hosts
- Rescheduling failed containers
- Networking
  - Containers on same host
  - Containers on different hosts
  - Container <-> Outside of the cluster
- Data sharing / persistence
- Security
  - Authentication
  - Authorization
  - Accounting



How do I know where to run a container?



vs



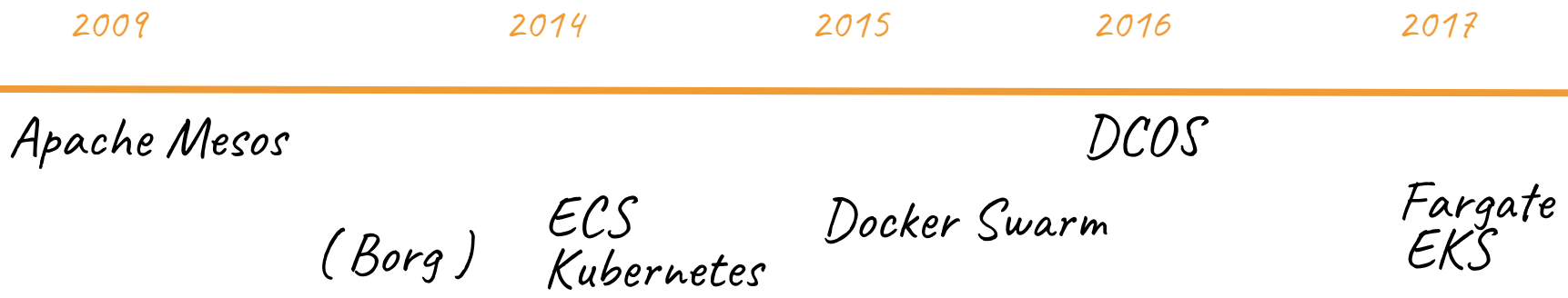
"Maximal density"

"Wherever"

Choices … so many choices …

Container orchestration is a common but not trivial problem

Multiple projects to address the same issue …



Elastic Container Service for Kubernetes (EKS)

EKS - Managed Kubernetes Service

Announced in re:Invent 2017, currently in preview in us-west-2

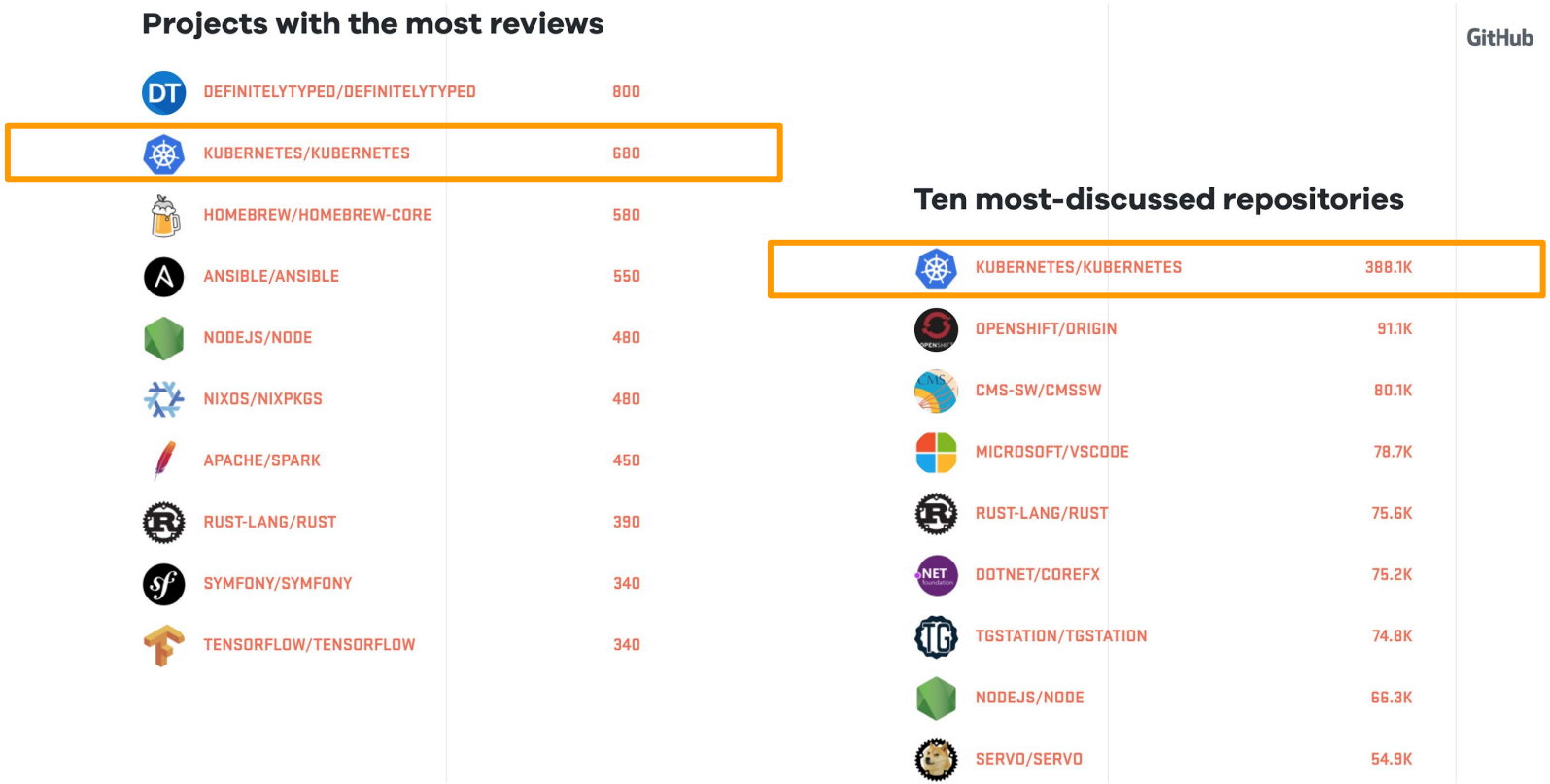
What is Kubernetes?

- “Helmsman” in Greek (κυβερνήτης)
- Also known as k8s or kube
- The most popular project on Cloud Native Computing Foundation (CNCF)
- March 6: First project that has reached “graduation” in CNCF

CLOUD NATIVE COMPUTING FOUNDATION

kubernetes

Kubernetes Popularity



Source: https://octoverse.github.com/

Why do People Run Kubernetes?

- Open Plugin Architecture. Runs everywhere
  - AWS
  - on-premise bare-metals
  - VMWare
  - other cloud providers
- Support for hybrid environments
  - Kube federation
- Kubernetes is popular
- Kubernetes is an interesting technology

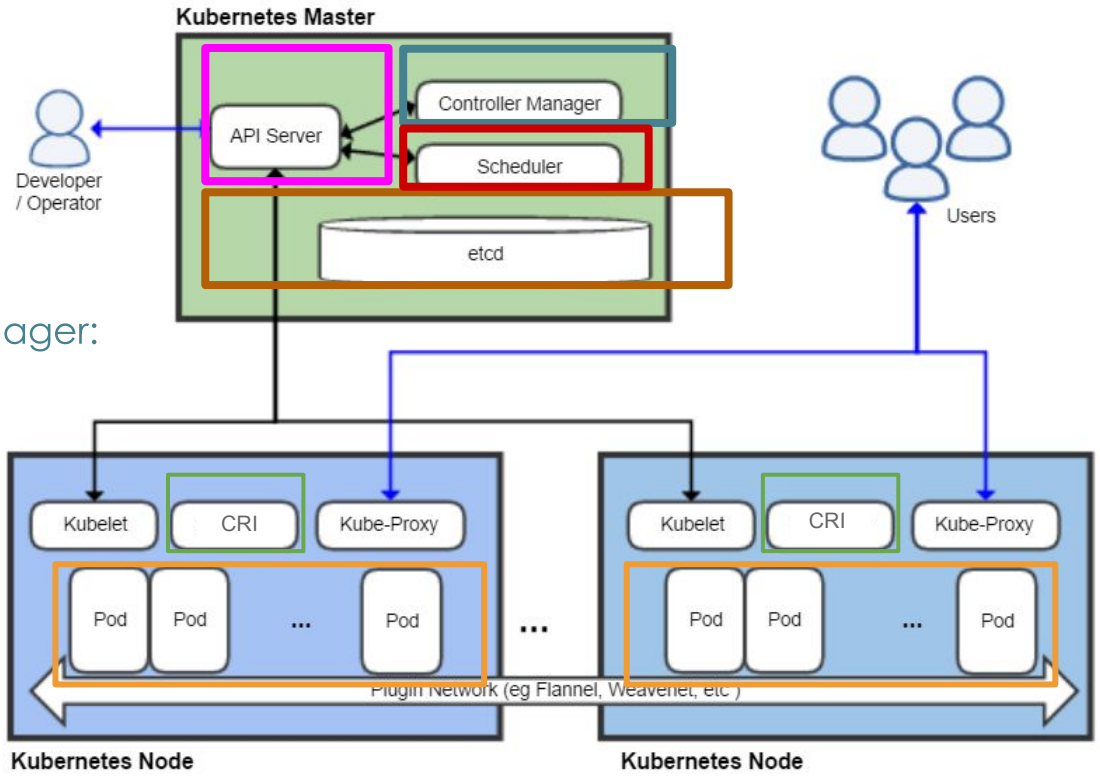
Kubernetes Masters and Workers

Kube-apiserver:
frontend of k8s
control plane

etcd: distributed
key-value store

Kube-controller-manager:
1+ process watch
for current state
and make changes
to bring to
desired state

Kube-scheduler:
Assign pods to
workers



Pod:
1+ containers
sharing the
same lifecycle

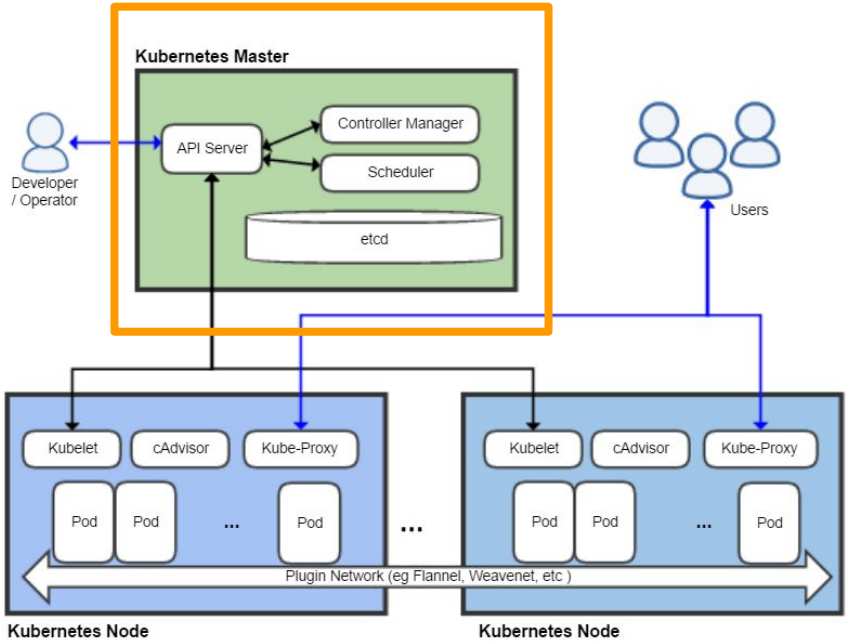
Container Runtime:
Docker (or rkt)

Source: https://en.wikipedia.org/wiki/Kubernetes#/media/File:Kubernetes.png

How “managed” is Elastic Container Service for Kubernetes (EKS)?

Managed **masters**

- Managed Control Plane
- Managed Persistence key-value store (i.e. etcd)



Advantages of EKS

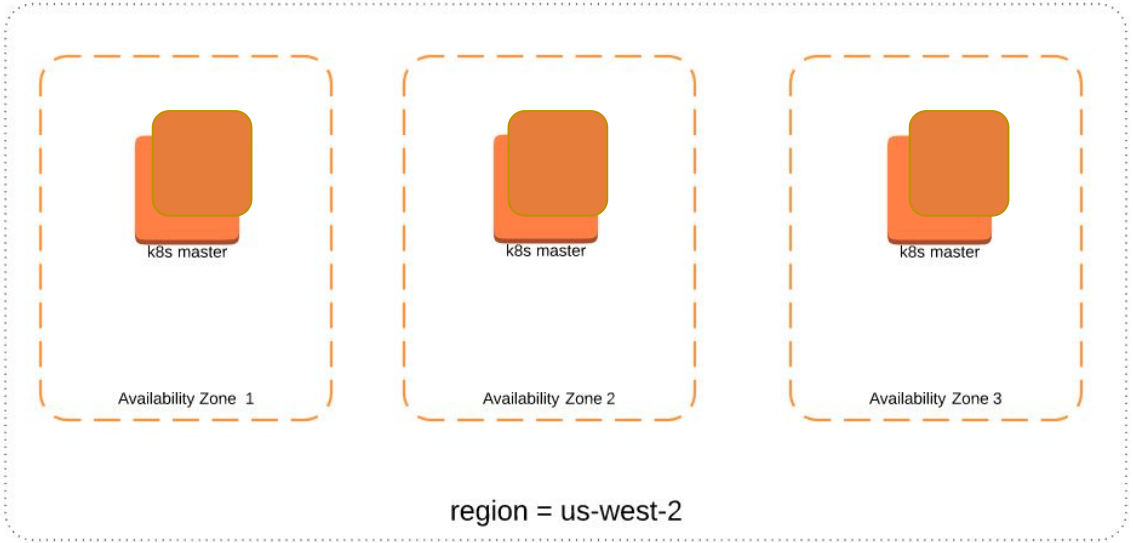
- Managed Kubernetes Control Plane
- Highly Available
- Autoscaling
- Automated Version Upgrade
- Integration with other AWS services
- Secure and Compliant

"Easiest and most compliant way to run Kubernetes on AWS!"

Highly Available + Autoscaling

3 Availables Zones:

- AZ = Isolated locations/facilities
- Multiple AZ = protection against datacenter failure
- Start off with 3 instances, if workload requires, autoscale up



Automated Version Update

K8s releases

- minor ~ every 3 months
- patch ~ every 2 weeks

Managing upgrade/patching = Full-time Job !!

EKS

- Automated patch release
- Scheduled minor release
  - latest + 2 earlier minor releases supported

Kubernetes Release	Date	Cadence
Christening of 1.0	10th July 2015	~one year from inception
From 1.0 to 1.1	9th November 2015	122 days
From 1.1 to 1.2	16th March 2016	128 days
From 1.2 to 1.3	1st July 2016	107 days
From 1.3 to 1.4	26th September 2016	87 days
From 1.4 to 1.5	12th December 2016	77 days
From 1.5 to 1.6	28th March 2017	106 days
From 1.6 to 1.7	30th June 2017	94 days
From 1.7 to 1.8	28th September 2017	90 days
From 1.8 to 1.9	15th December 2017	78 days
From 1.9 to 1.10	ETA mid-March 2018	~100 days

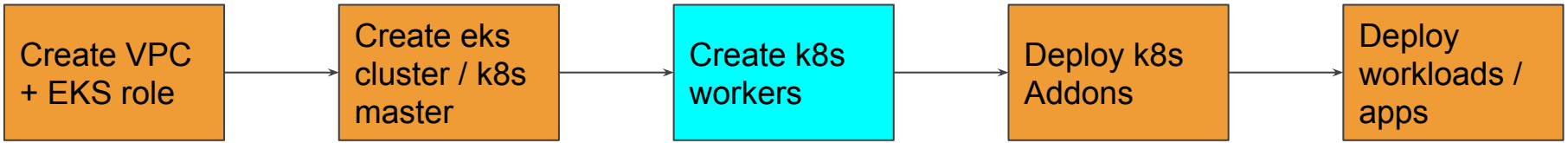
Source: https://gravitational.com/blog/kubernetes-release-cycle/

Integration with other AWS services

Authentication	IAM
Network Isolation	VPC
Load Balancing	ELB (ALB + NLB on the pike)
Pod-Networking	ENI
Private Network Access	AWS Private Link
Logging/Auditing	CloudWatch Logs/CloudTrail

EKS = Secure and Compliant K8S

- HIPAA and PCI compliant
- Security already built in
  - Network Policies (Calico / Tigera)
  - IAM Authentication (heptio authenticator for AWS)
- Best practice to deploy k8s workers
  - CloudFormation blueprint to deploy workers
  - AMI baseline to provision workers
  - Security side by side to Flexibility



Kubernetes in the News

DevOps

Kubernetes bug ate my banking app! How code flaw crashed Brit upstart

Monzo engineering chief details exact cause of outage

By Thomas Claburn in San Francisco 31 Oct 2017 at 19:51 32 SHARE

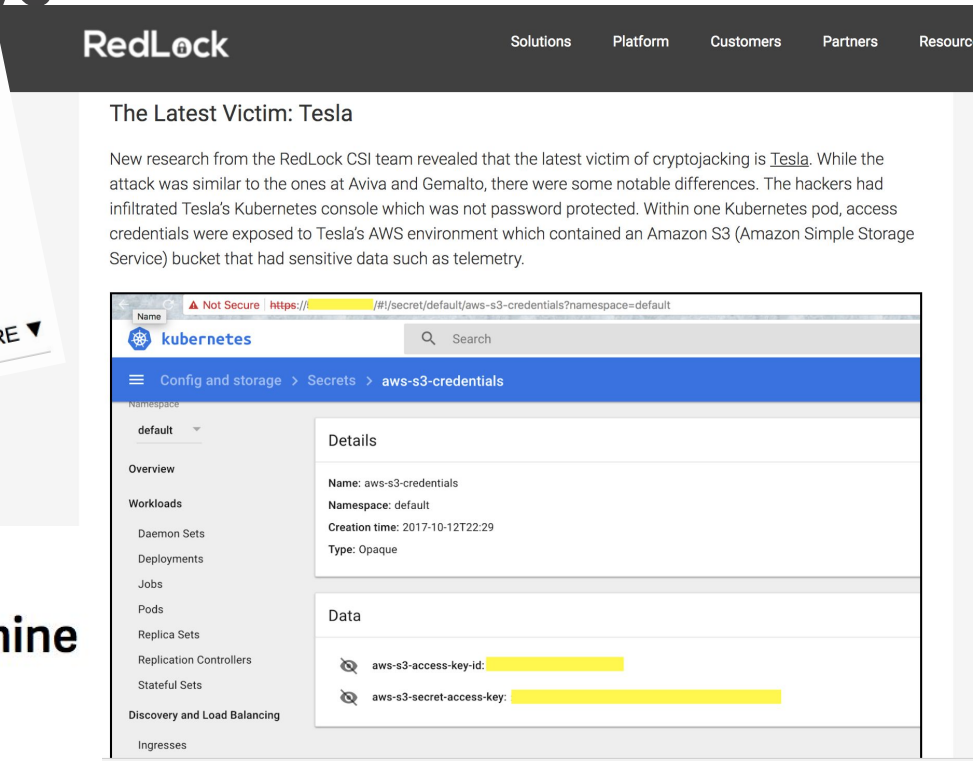
TRANSPORTATION | CARS | TESLA

Tesla's cloud was used by hackers to mine cryptocurrency

Mining bitcoin on Elon's dime

By Andrew J. Hawkins | @andyjayhawk | Feb 20, 2018, 1:39pm EST

SHARE



RedLock Solutions Platform Customers Partners Resourc

The Latest Victim: Tesla

New research from the RedLock CSI team revealed that the latest victim of cryptojacking is Tesla. While the attack was similar to the ones at Aviva and Gemalto, there were some notable differences. The hackers had infiltrated Tesla's Kubernetes console which was not password protected. Within one Kubernetes pod, access credentials were exposed to Tesla's AWS environment which contained an Amazon S3 (Amazon Simple Storage Service) bucket that had sensitive data such as telemetry.

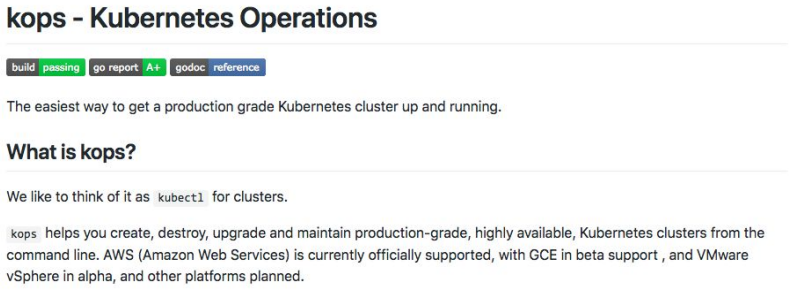
Source: https://blog.redlock.io/cryptojacking-tesla
https://community.monzo.com/t/resolved-current-account-payments-may-fail-major-outage-27-10-2017/26296/94
https://www.theverge.com/2018/2/20/17032684/tesla-cloud-hacker-cryptocurrency-redlock

EKS Demo

EKS = Best Agnostic Container Orchestrator

Before EKS

- Kops
- Heptio Quickstart
  https://aws.amazon.com/quickstart/architecture/heptio-kubernetes/



kops - Kubernetes Operations

build passing | go report A+ | godoc reference

The easiest way to get a production grade Kubernetes cluster up and running.

What is kops?

We like to think of it as `kubectl` for clusters.

`kops` helps you create, destroy, upgrade and maintain production-grade, highly available, Kubernetes clusters from the command line. AWS (Amazon Web Services) is currently officially supported, with GCE in beta support , and VMware vSphere in alpha, and other platforms planned.

Issue

"Operating Kubernetes is not for the faint of heart!"

Need agnostic container orchestration on AWS?
Use EKS !!

AWS Fargate

AWS Lambda - A Known Model

	AWS Lambda
Run	Code / Function
Resources	CPU and Memory
Pricing unit	100ms
Management	No Idle Capacity Highly Available Elastic Minimal overhead
Use case	Event driven programming Scheduled Job

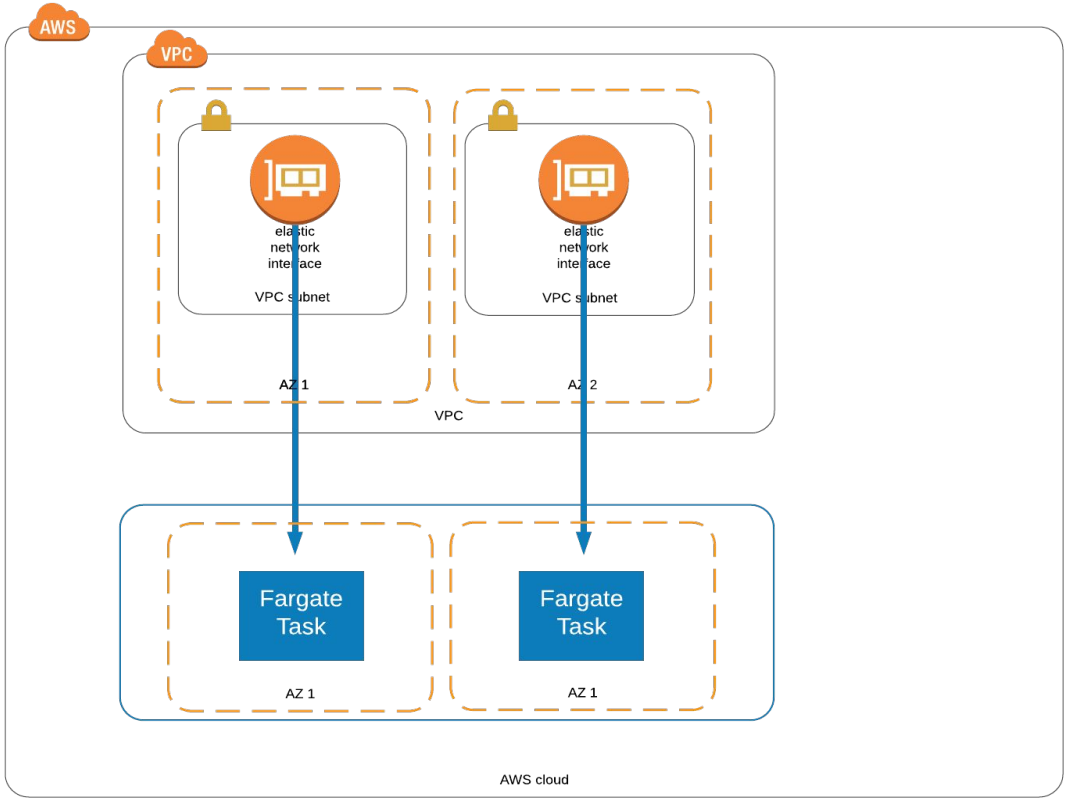
AWS Fargate - Quite Similar!

	AWS Lambda	AWS Fargate
Run	Code / Function	Container
Resources	CPU and Memory	CPU and Memory
Pricing unit	100ms	1s
Management	No Idle Capacity Highly Available Elastic Minimal ops overhead	No Idle Capacity Highly Available Elastic Minimal ops overhead
Use case	Event driven programming Scheduled Job	Scheduled Job

Running in Shared storage Securely

- Tasks run in shared resources
- Max layer size of image = 4 GB
- Container storage = 10GB
- Size of shared volume shared by containers within same task = 4 GB

- ENI is directly attached to task to ensure security group / firewall setting



Fargate Definitions

Same as
ECS Task Definitions

- portMappings
- image
- logConfiguration

Special fields

- networkMode
- compatibilities

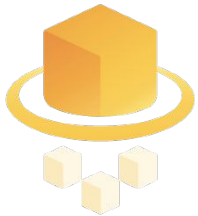
```
{
  "containerDefinitions": [
    {
      "memory": 128,
      "networkMode": "awsvpc",
      "portMappings": [
        {
          "hostPort": 80,
          "containerPort": 80,
          "protocol": "tcp"
        }
      ],
      "essential": true,
      "name": "nginx-container",
      "image": "nginx",
      "logConfiguration": {
        "logDriver": "awslogs",
        "options": {
          "awslogs-group": "/ecs/mency-fargate-nginx",
          "awslogs-region": "us-east-1"
        }
      },
      "cpu": 0
    }
  ],
  "compatibilities": : [
        "EC2",
        "FARGATE"
      ],
  "family": "mency-fargate-nginx"
}
```

Fargate Demo

Fargate

Serverless Container Orchestration

- Fargate on ECS
  - already publicly available in us-east-1
- Fargate on EKS
  - in 2018



AWS Fargate

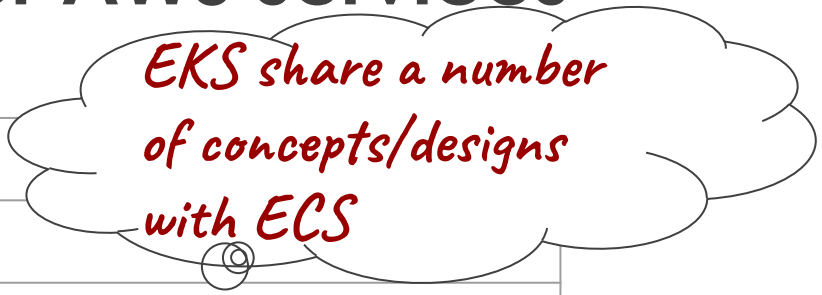
Quick/sporadic container workload?
Want to offload all ops? Use Fargate!!

Elastic Container Service (ECS)

Elastic Container Service

- Introduced in November 2014
- Native AWS container orchestration
- Tight integration with other AWS services

ECS integration with other AWS services



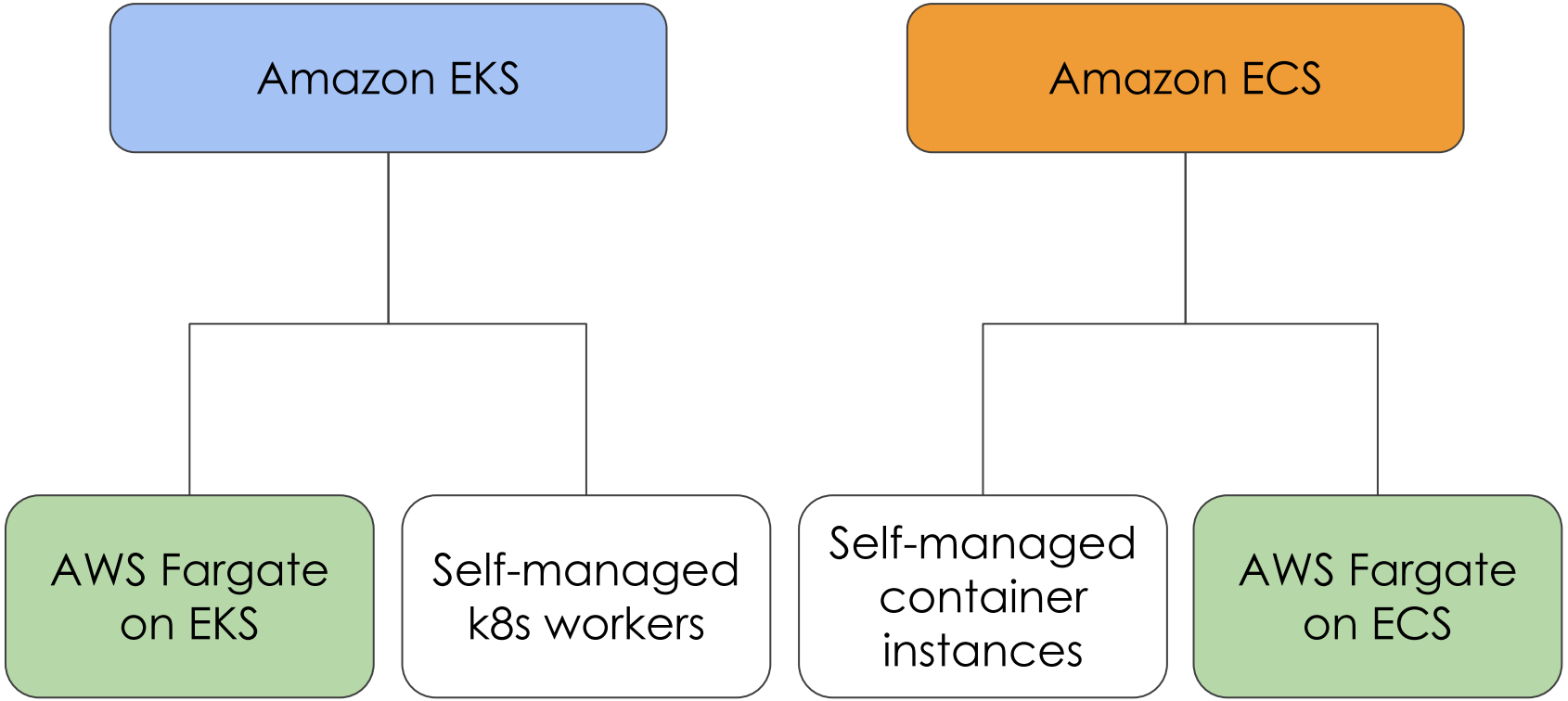
Authentication/Authorization	IAM
Network Isolation	VPC
Load Balancing	ALB + NLB
Task networking	ENI (awsvpc mode)
Monitoring and Task Scaling	CloudWatch
Image Repository	Elastic Container Repository (ECR)
Host Scaling	CloudWatch + Autoscaling Groups
Logging / Auditing	CloudWatch Logs / CloudTrail

ECS satisfies the more advanced workload

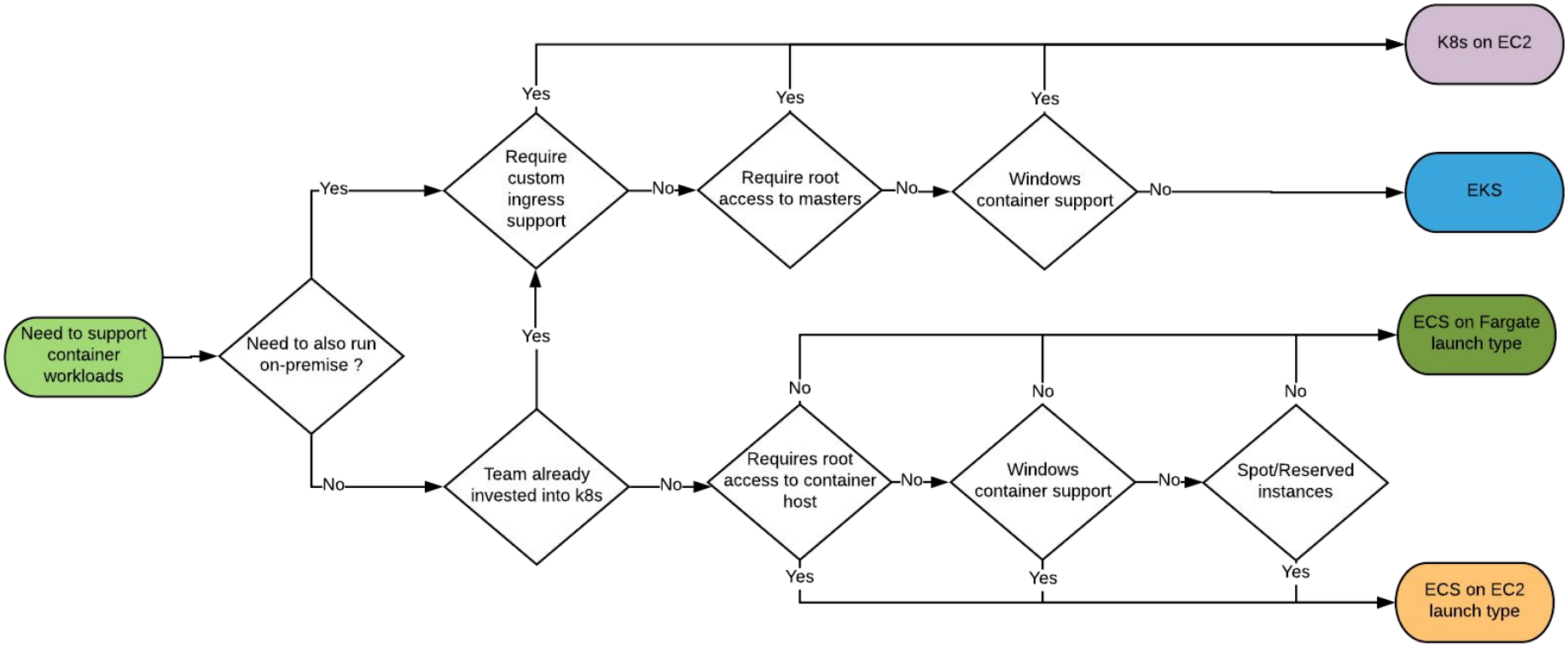
- Spot instances
- Windows containers workload
- GPU support
- Root access to container instances required
- Custom “worker” instances
  - Key requirement: ECS container agent
  - Public AMI’s / baseline available on
    - Amazon Linux
    - Ubuntu
    - Windows
    - CoreOS

Picking between
EKS, ECS, Fargate and K8s on EC2

The services are not mutually exclusive



How to pick


Need to support container workloads
Need to also run on-premise ?
Yes
No
Require custom ingress support
Yes
No
Require root access to masters
Yes
No
Windows container support
Yes
No
K8s on EKS
EKS
Team already invested into k8s
Yes
No
Requires root access to container host
No
Yes
Windows container support
No
Yes
Spot/Reserved instances
No
Yes
ECS on Fargate launch type
ECS on EC2 launch type

Thank you!