Running Docker clusters on AWS

Julien Simon, Principal Technical Evangelist
julsimon@amazon.fr
@julsimon

The problem

Given a certain amount of processing power and memory,

how can we best manage an arbitrary number of apps running in Docker containers?

http://tidalseven.com

Docker on Amazon Web Services

Amazon EC2 Container Service (ECS)

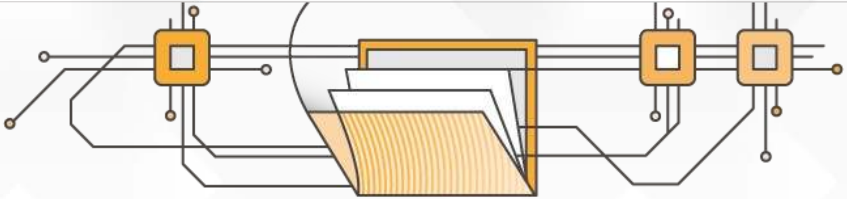
- https://aws.amazon.com/ecs/
- Launched in 04/2015
- Available in eu-west-1 & eu-central-1
- No additional charge

Amazon EC2 Container Registry (ECR)

- https://aws.amazon.com/ecr/
- Launched in 12/2015
- Available in eu-west-1
- Free tier: 500MB / month for a year
- $0.10 / GB / month + outgoing traffic

amazon
web services

Breaking news ;)



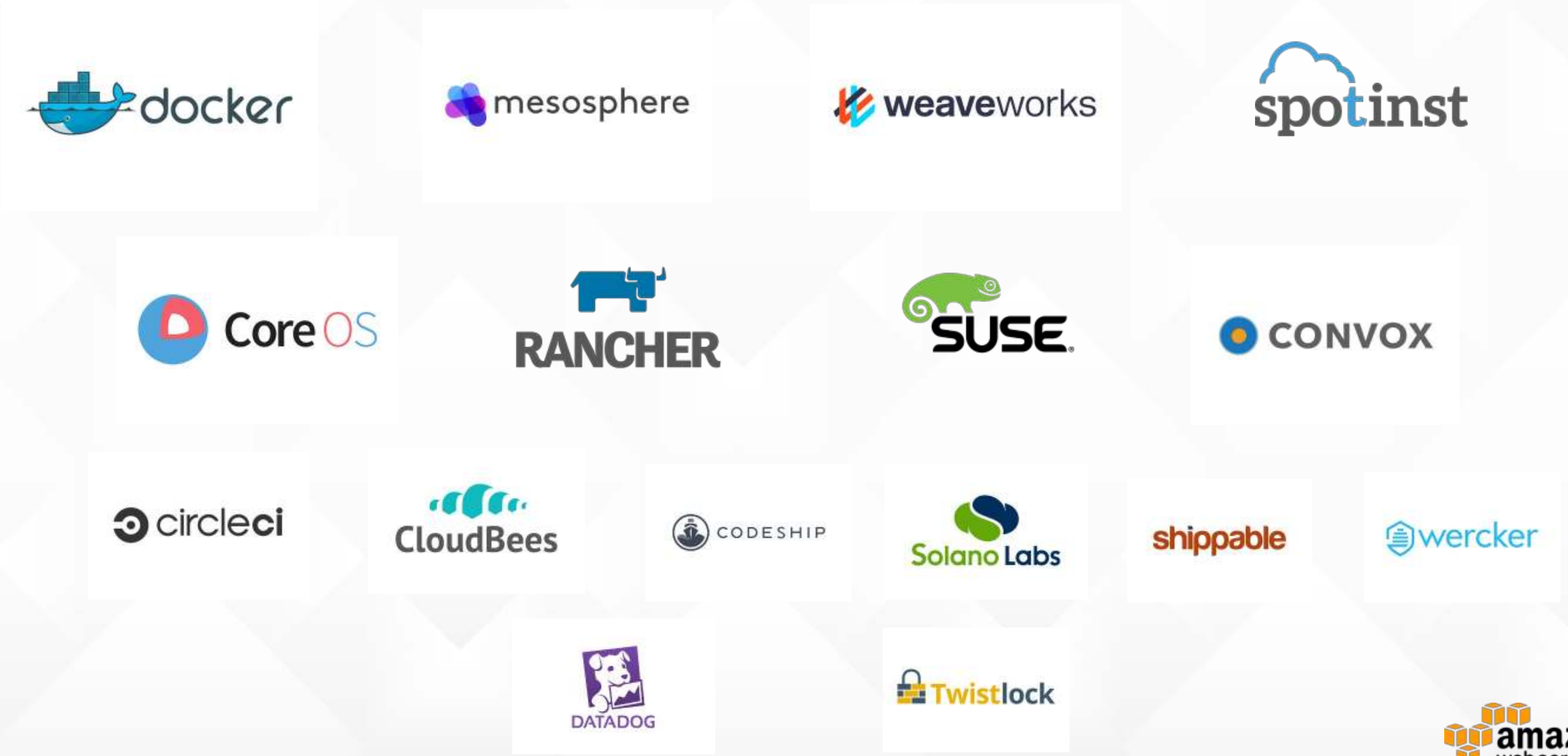
Amazon Elastic File System (EFS)

- https://aws.amazon.com/efs/
- Launched June 29th
- NFSv4.1 server mountable by EC2 instances
- Allows shared storage for EC2 instances (Docker volumes anyone?)
- Scales capacity automatically and instantly as you add or remove files
- Fully managed service

- Available in eu-west-1 : $0.33 / GB / month

AWS Partners

https://aws.amazon.com/fr/containers/partners/



Case studies

Case study: Coursera

https://www.youtube.com/watch?v=a45J6xAGUvA

Coursera deliver Massive Open Online Courses (14 million students, 1000+ courses). Their platform runs a large number of batch jobs, notably to grade programming assignments. Grading jobs need to run in near-real time while preventing execution of untrusted code inside the Coursera platform.

After trying out some other Docker solutions, Coursera have picked Amazon ECS and have even written their own scheduler.

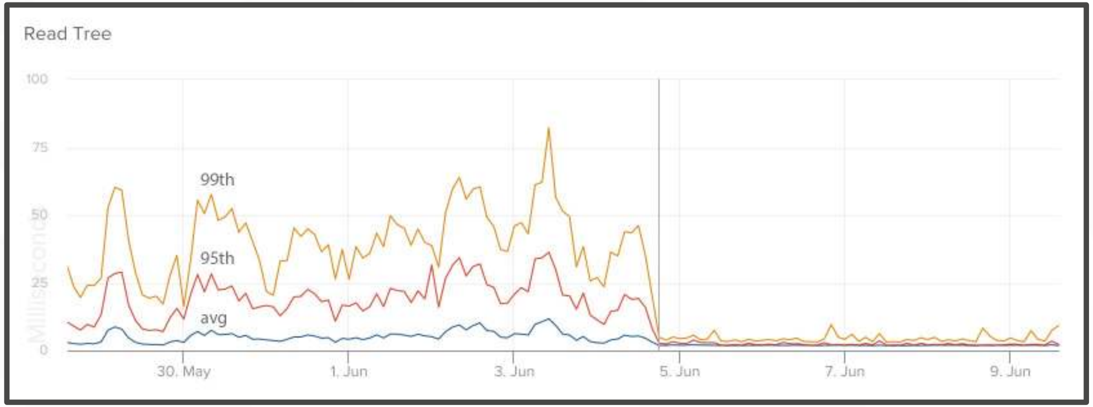
“*Amazon ECS enabled Coursera to focus on releasing new software rather than spending time managing clusters*” - Frank Chen, Software Engineer

Case study: Remind

https://www.youtube.com/watch?v=8zbbQkszP04

Messaging platform for teachers, parents and students (35M users, 2.5B messages, 50% of U.S. public schools)

Micro-service platform deployed on Heroku, migrated to Amazon ECS (36 nodes in Q4'15)



"*Moving to Amazon ECS significantly improved our service performance*" Jason Fischl, VP of Engineering

Case study: Hailo

https://aws.amazon.com/fr/solutions/case-studies/hailo/

Hailo allows people to hail licensed taxis directly to their location (60,000+ drivers, 1M+ passengers).

Hailo has evolved from a monolithic application running in one AWS region to a microservice-based architecture running across multiple regions.

Hailo decided to schedule containers based on service priority and other runtime metrics atop an elastic resource pool. They chose Amazon ECS as the cluster manager because it is a managed service that can easily enforce task state and fully exposes the cluster state via API calls: http://fr.slideshare.net/nathariel/microservices-and-elastic-resource-pools-with-amazon-ec2-container-service

Case study: Segment

Segment

https://aws.amazon.com/fr/solutions/case-studies/segment/

Segment provides a service used by businesses to collect customer data for later use in analytics and marketing.

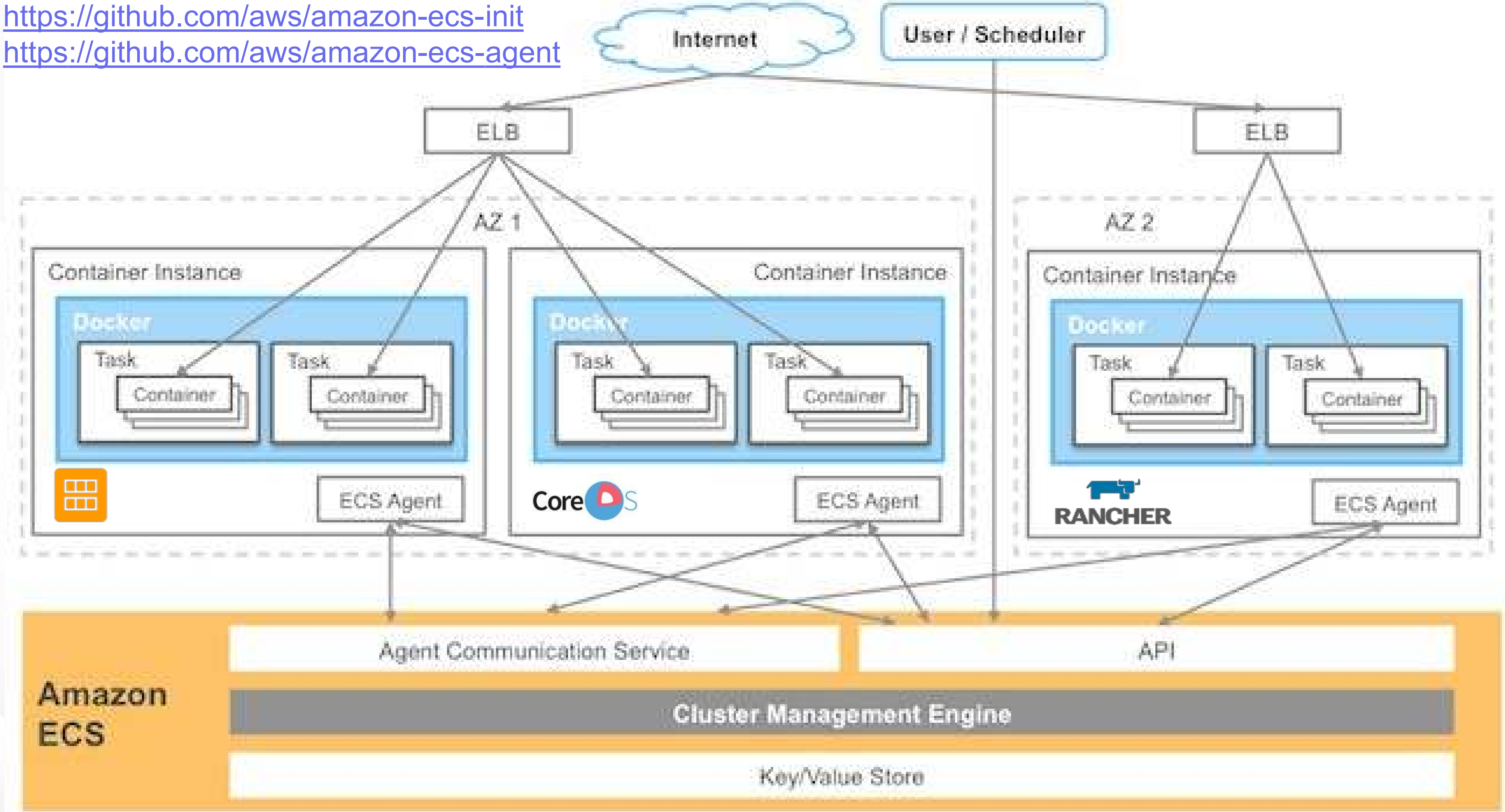
Different micro-services such as API, CDN, and App are deployed on different Amazon ECS clusters. Each service registers to an ELB and Amazon Route 53 points a local entry at each ELB. Services can communicate with each other through DNS.

Segment have also built their own PaaS on top of AWS: https://segment.com/blog/the-segment-aws-stack/

> “*Switching to Amazon ECS has greatly simplified running a service without needing to worry about provisioning or availability*”
> Calvin French-Owen, Cofounder and CTO

Architecture

https://github.com/aws/amazon-ecs-init
https://github.com/aws/amazon-ecs-agent



http://www.allthingsdistributed.com/2015/07/under-the-hood-of-the-amazon-ec2-container-service.html

The Amazon ECS CLI in one slide

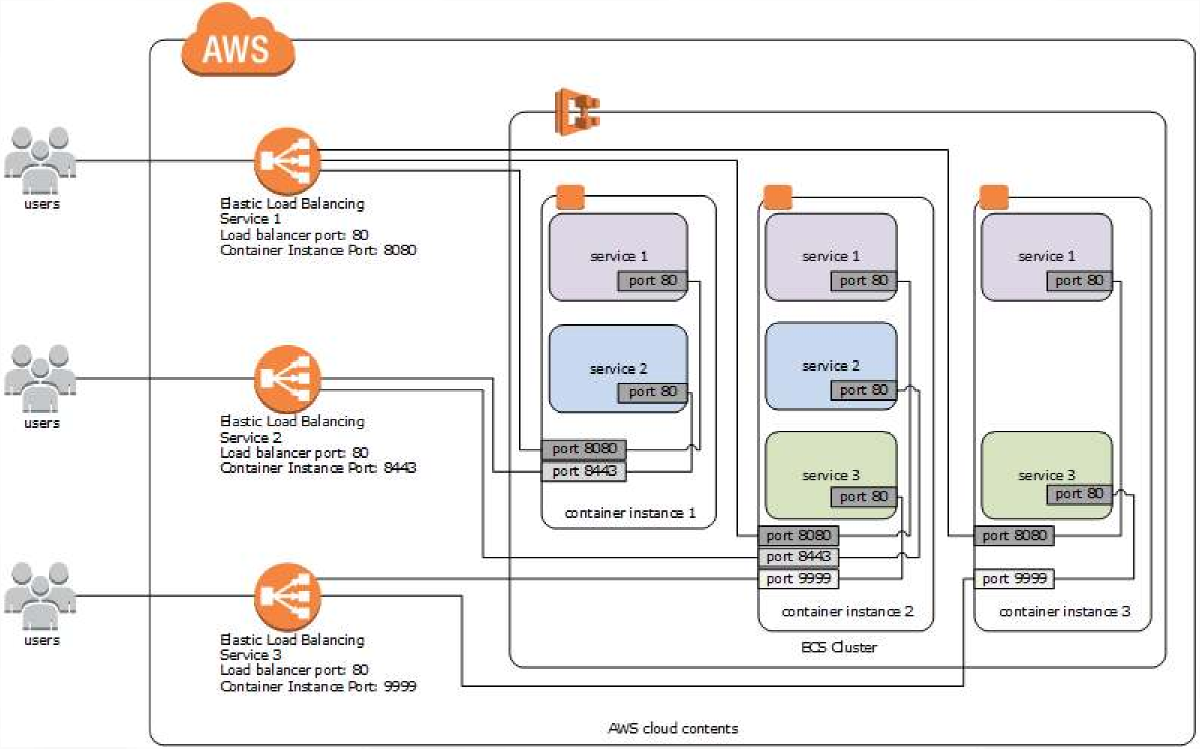
https://github.com/aws/amazon-ecs-cli

```
ecs-cli configure --cluster myCluster --region eu-west-1
ecs-cli up --keypair myKey --capability-iam –size 3
ecs-cli down myCluster

ecs-cli compose service up
ecs-cli compose service ps
ecs-cli compose service scale 8
ecs-cli compose service stop
ecs-cli compose service delete

aws ecs list-clusters
aws ecs describe-clusters --cluster myCluster
aws ecs list-container-instances --cluster myCluster
```

DEMO #1

RancherOS on Amazon ECS
+ Rancher Server

Fixed ports, ELB

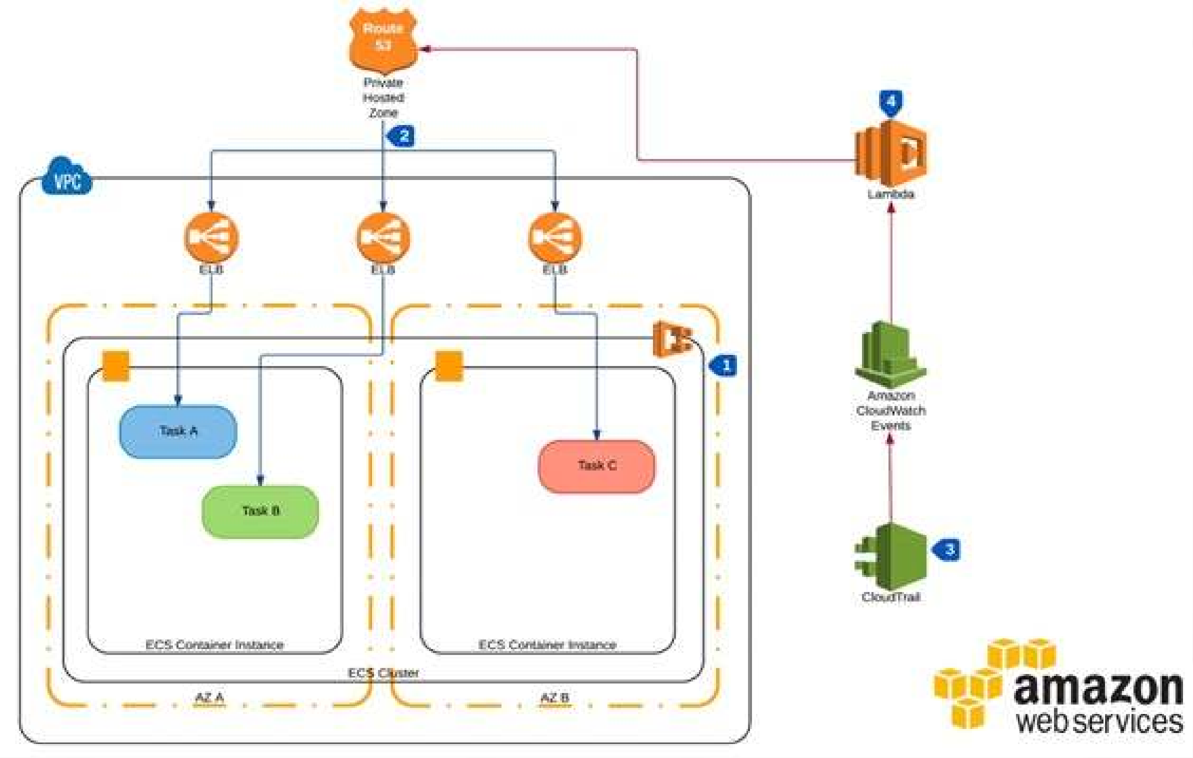


- No service discovery: configuration required (env. variables for ELBs)
- Only 1 container from a given image per ECS instance

Amazon EC2 Container Service Developer Guide

Fixed ports, ELB, DNS

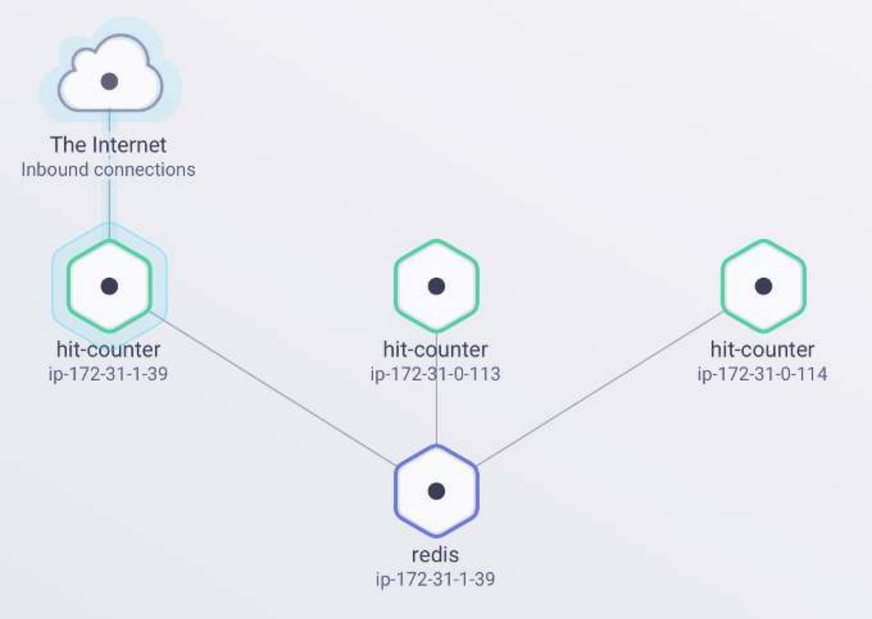
https://aws.amazon.com/blogs/compute/service-discovery-an-amazon-ecs-reference-architecture/



- Service registration done in Route 53 by CloudWatch Events and Lambda (new CNAME for the ELB)
- Service discovery with DNS
- Only 1 container from a given image per ECS instance

Fixed ports, Weave, DNS

https://aws.amazon.com/blogs/apn/architecting-microservices-using-weave-net-and-amazon-ec2-container-service/



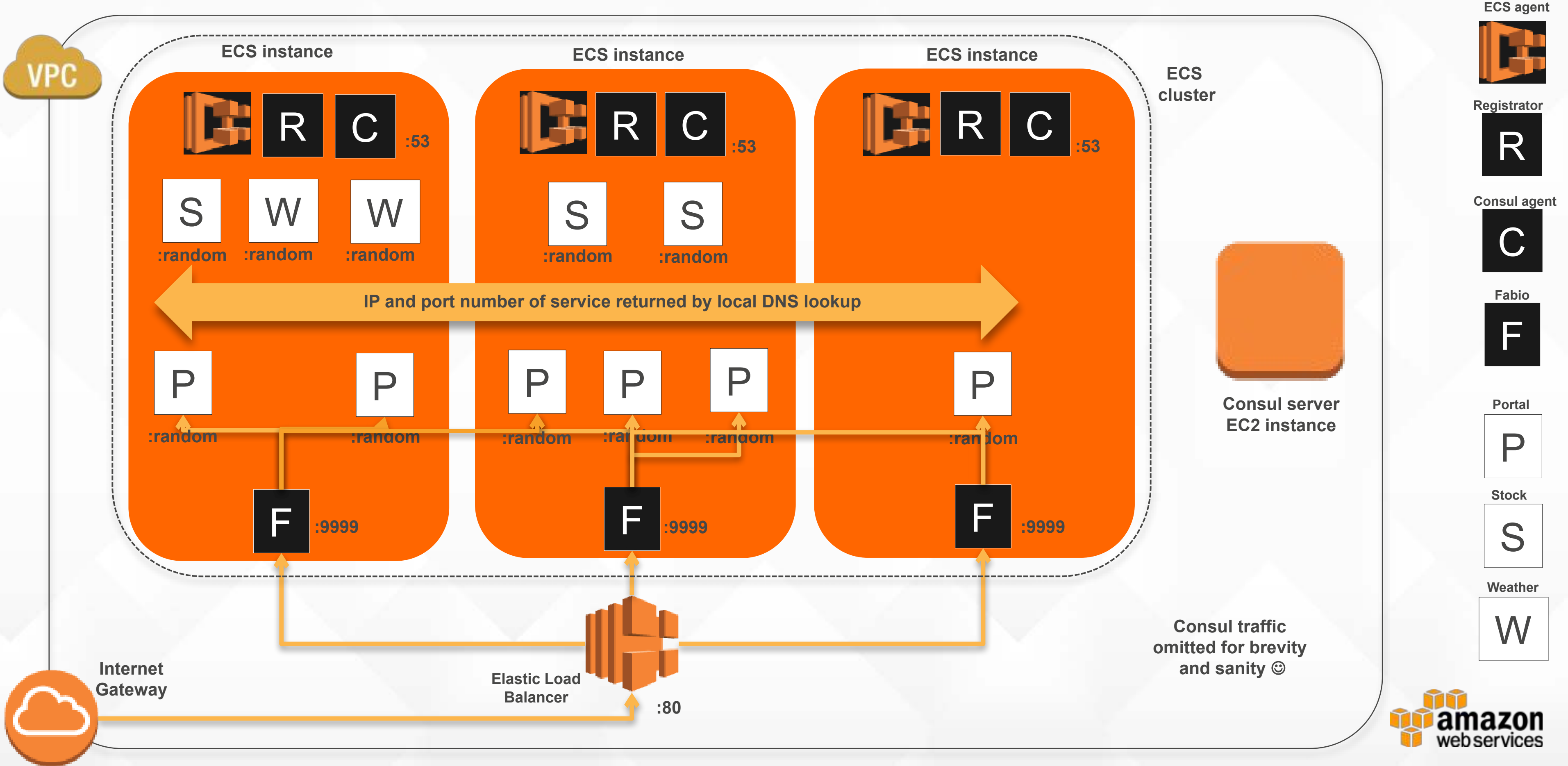
- Service registration done by Weave
  - gossip protocol, no central server
  - IP address only
- Service discovery & load balancing done with DNS
- Still need an ELB for Internet-facing services
- Only 1 container from a given image per ECS instance

DEMO #2

Amazon Linux on Amazon ECS
+ Weave

Random ports, Registrator, Fabio, Consul

https://aws.amazon.com/blogs/compute/service-discovery-via-consul-with-amazon-ecs/ + tweaks ;)



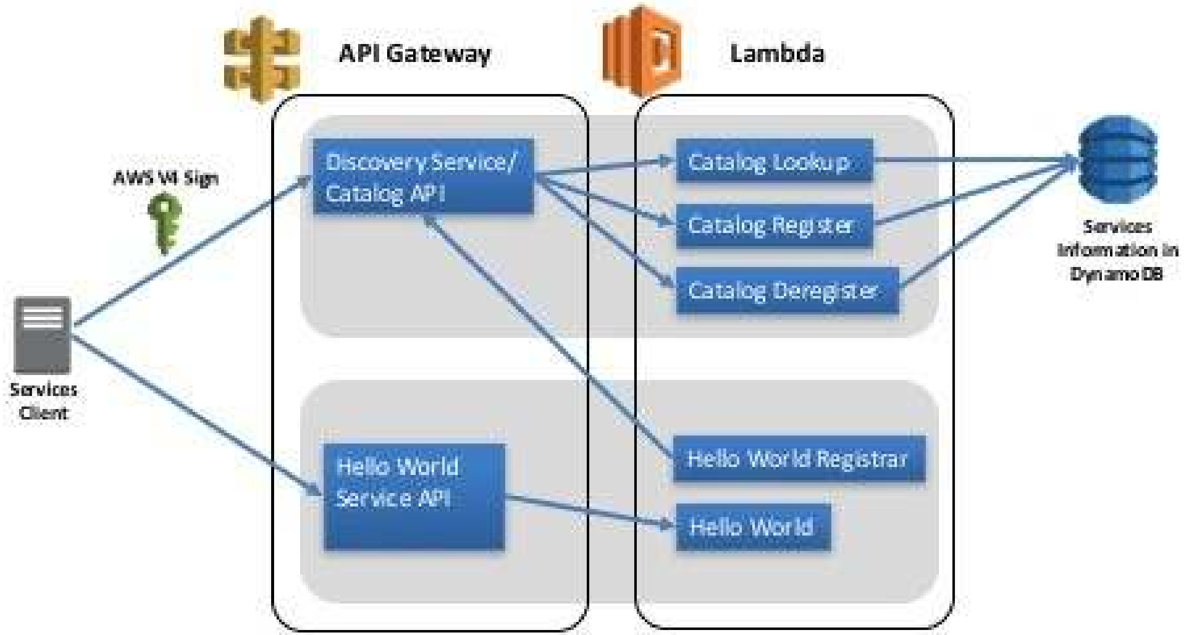
Consul traffic omitted for brevity and sanity ☺

DEMO #3

Amazon Linux on Amazon ECS
+ Registrator + Consul + Fabio

Another option: serverless micro-services

https://aws.amazon.com/blogs/developer/serverless-service-discovery-part-1-get-started/



Further reading

Tech articles by Werner Vogels, CTO, Amazon.com

http://www.allthingsdistributed.com/2014/11/amazon-ec2-container-service.html

http://www.allthingsdistributed.com/2015/04/state-management-and-scheduling-with-ecs.html

http://www.allthingsdistributed.com/2015/07/under-the-hood-of-the-amazon-ec2-container-service.html

Amazon ECS videos @ AWS re:Invent 2015

Amazon ECS: Distributed Applications at Scale https://www.youtube.com/watch?v=eun8CqGqdk8

Turbocharge Your Deployment Pipeline with Containers https://www.youtube.com/watch?v=o4w8opVCI-Q

From Local Docker Development to Production https://www.youtube.com/watch?v=7CZFpHUPqXw

3rd party software

https://www.weave.works

https://www.consul.io/

https://github.com/eBay/fabio

AWS User Groups

Lille
Paris
Rennes
Nantes
Bordeaux
Lyon
Montpellier
Toulouse

facebook.com/groups/AWSFrance/

@aws_actus

Thank you!

Julien Simon, Principal Technical Evangelist
julsimon@amazon.fr
@julsimon