

A microservice architecture with Docker

Arnaud Cogoluègues, Zenika

Context

(health) insurance company

existing web applications

monolithic relational database

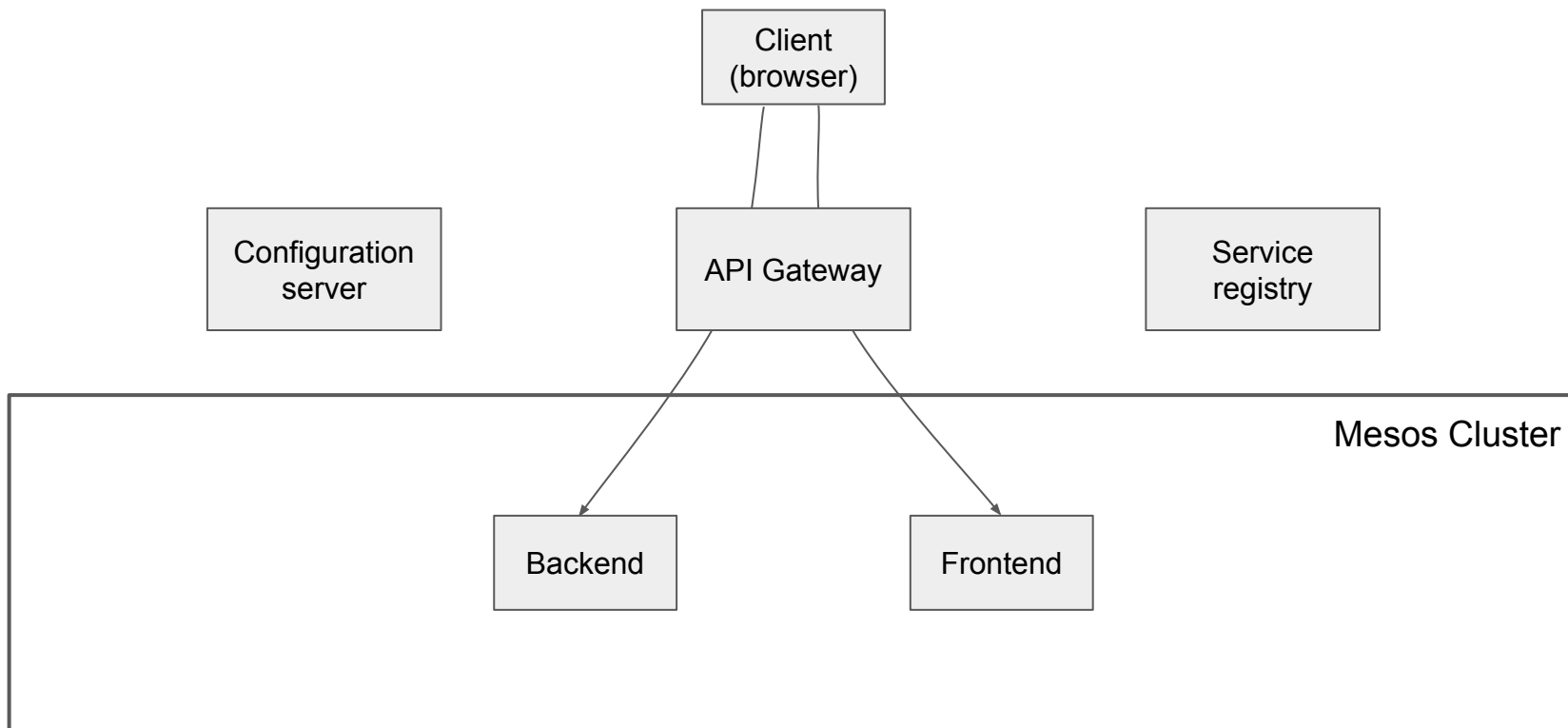
most of the business logic in the database

Context

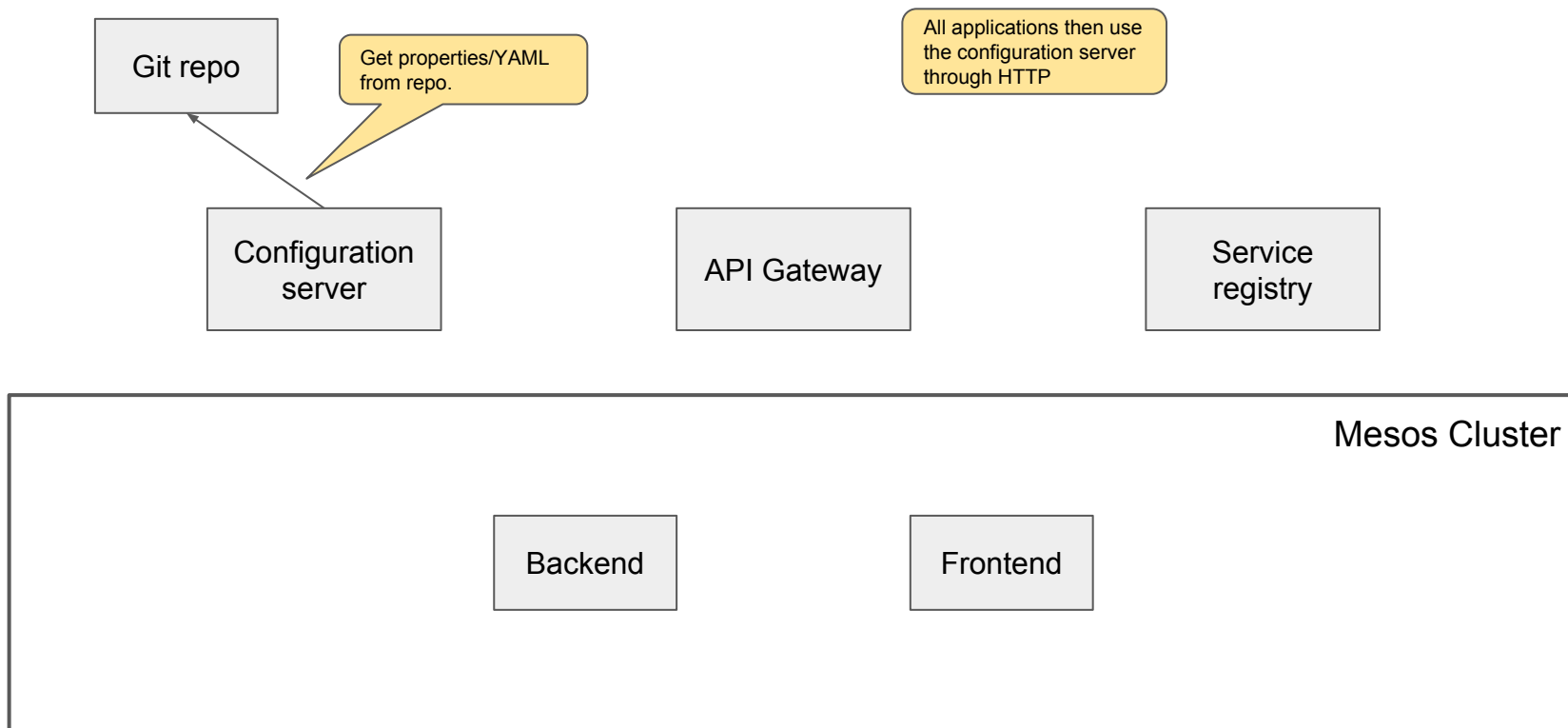
problem: difficult to re-use business logic

solution: expose high business value web services

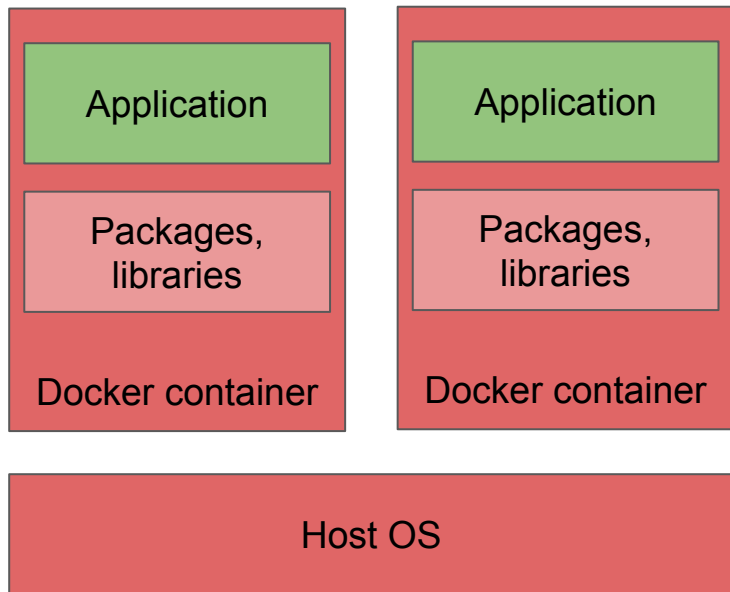
Big picture



Configuration server



Docker primer

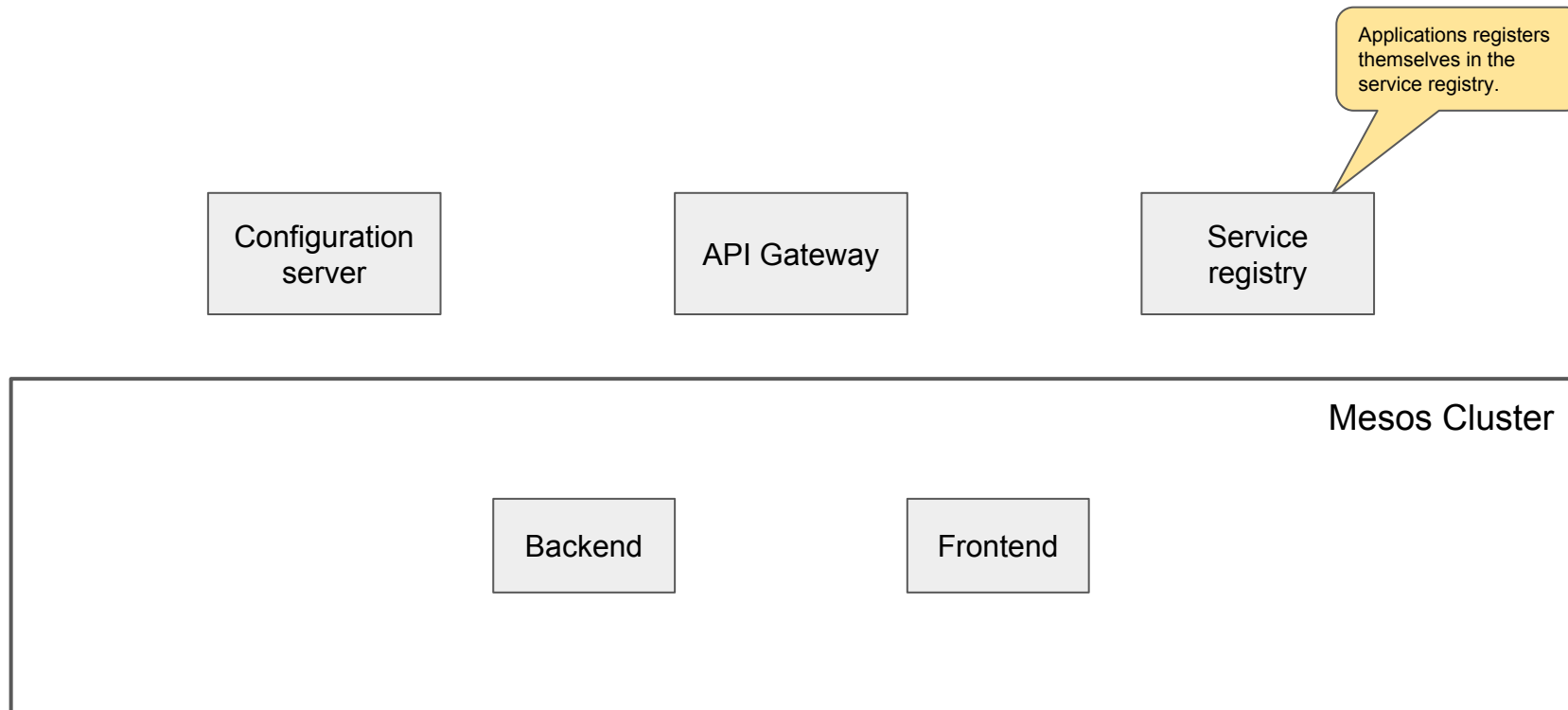


Binary isolation

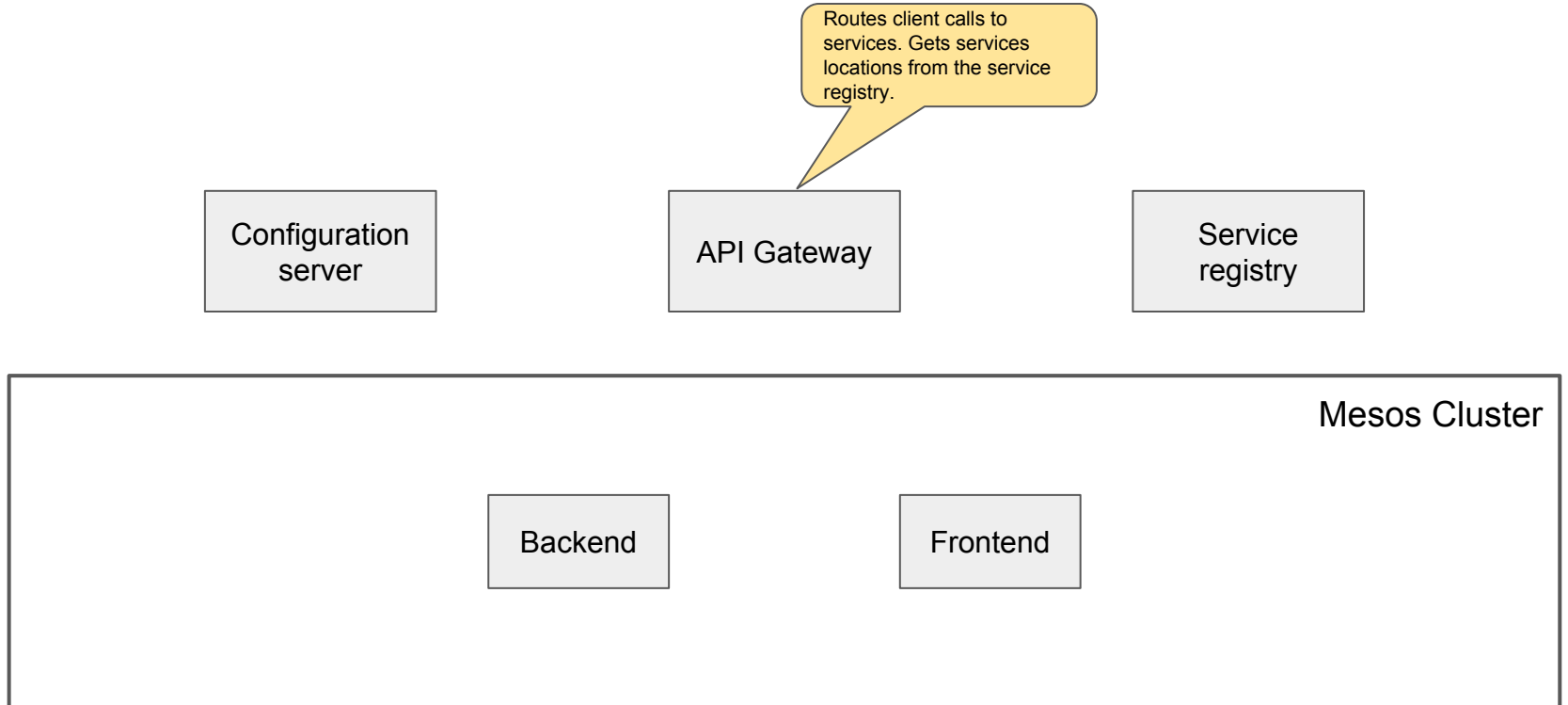
Resource isolation

Unified packaging

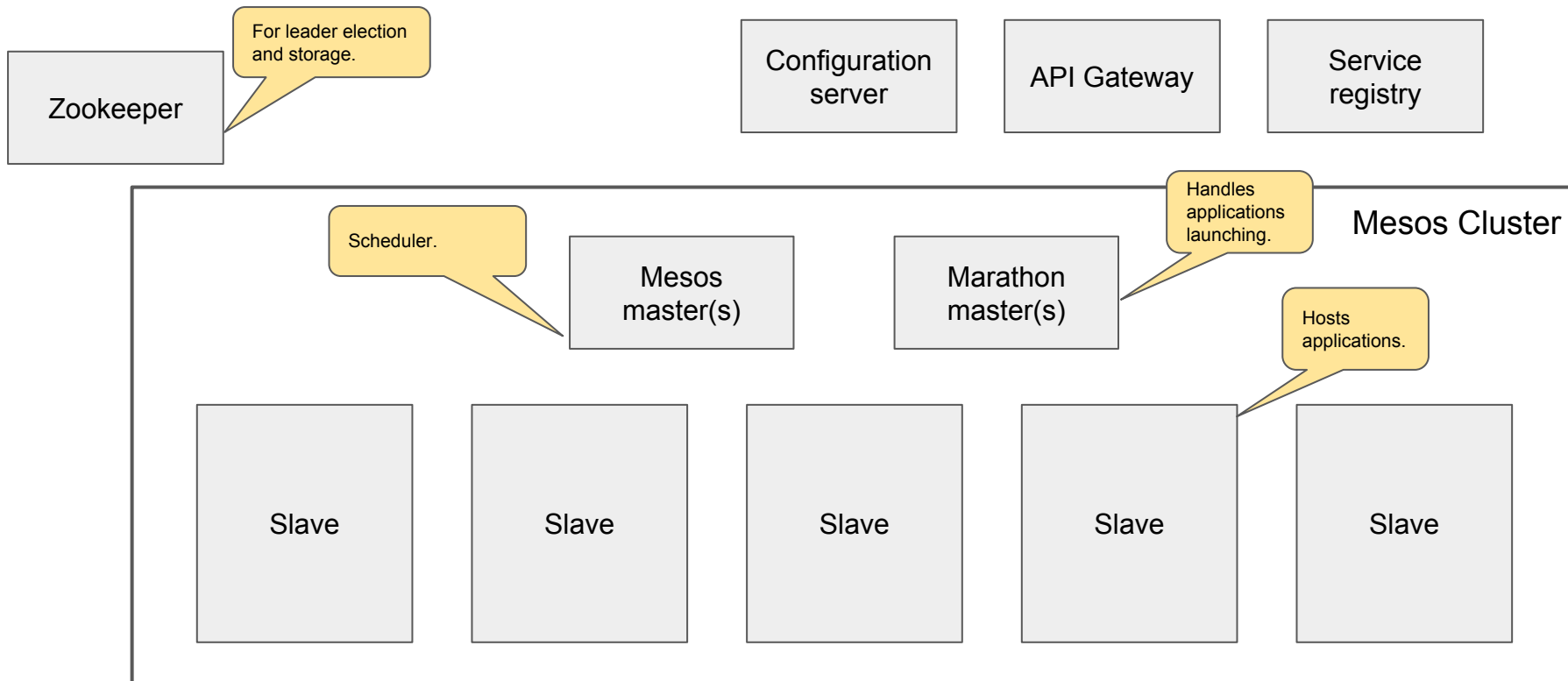
Service registry



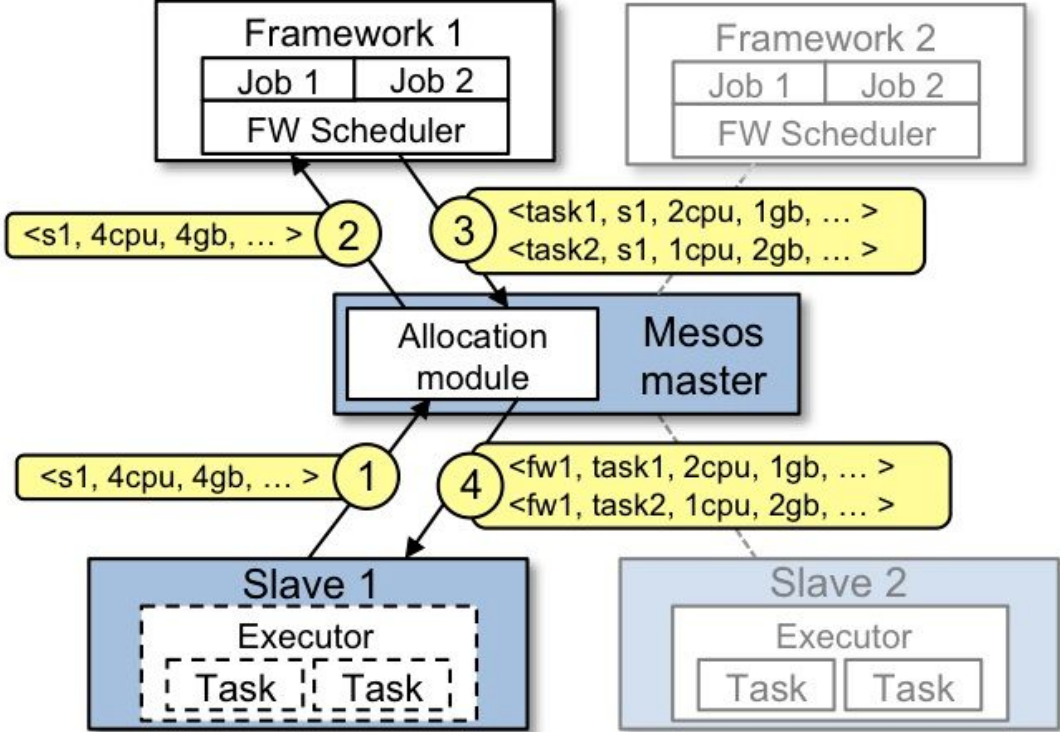
API gateway



Mesos cluster



Mesos 2-tier resource scheduling



Multiple environments

Keep configuration server and Mesos cluster.
Add API Gateway et Service Registry instances for each environment.

Configuration server

API Gateway (QA)

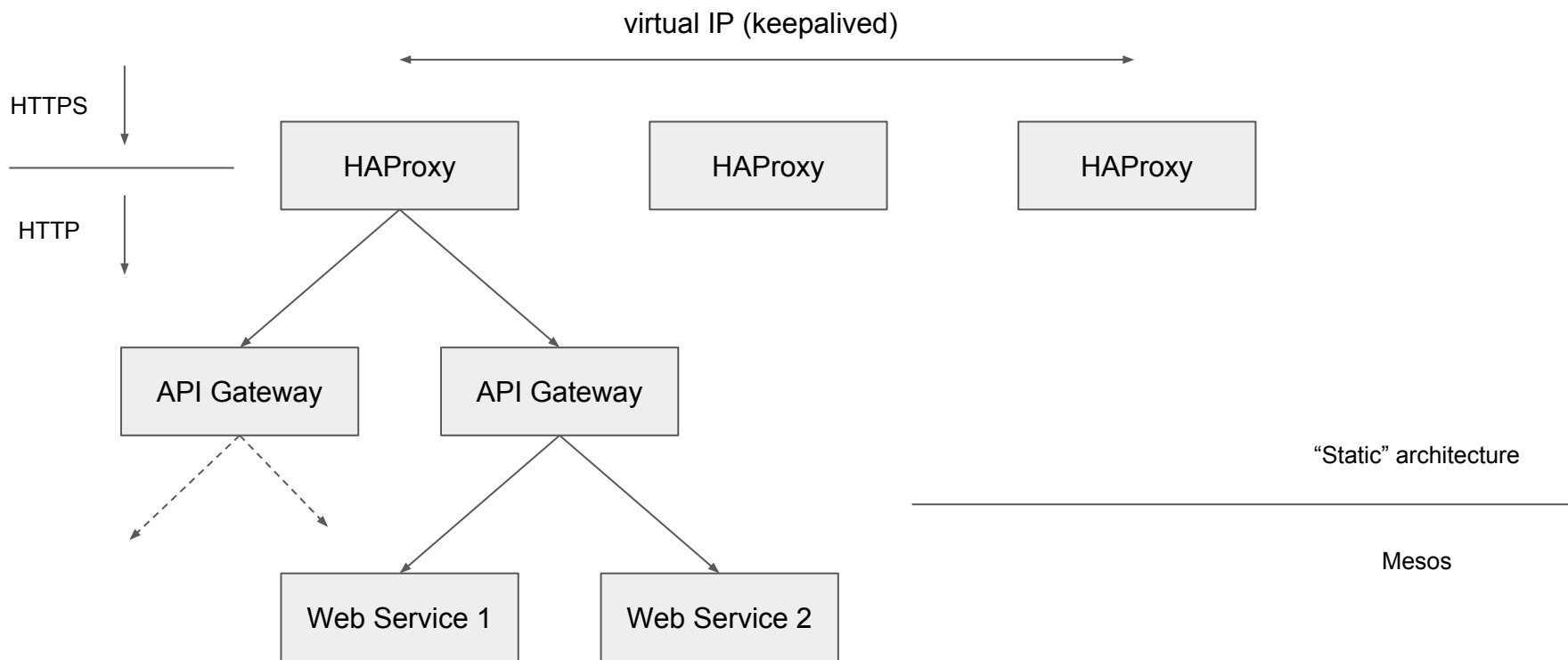
Service registry (QA)

API Gateway (production)

Service registry (production)

Mesos Cluster

Architecture



Tools

What	Tool	Alternatives
OS	CentOS, Debian	Ubuntu, Fedora, CoreOS, Rancher OS
Configuration Management	Ansible	Chef, Puppet, Salt
Configuration Server	Spring Cloud Configuration Server	etcd, Consul, Zookeeper, Netflix Archaius
Service Discovery	Netflix Eureka	Consul, etcd, Zookeeper
Dynamic Load Balancing	Netflix Zuul	HAProxy, Traefik
Orchestration	Mesos & Marathon	Rancher, Fleet, Kubernetes
Centralized Logging	Graylog	ELK

Disruptive?

stateless: no web session, no state on Mesos slaves

logs: no file, GELF + UDP

backup: none for Mesos masters and slaves

redundancy: no hardware redundancy for Mesos slaves

environments: dev and production apps on same hosts

Not so disruptive

OS: nothing fancy

Docker: only for stateless applications

storage: in traditional VM, not in container

Docker in production

One packaging to rull them all

Hard to measure daemon memory consumption

Use direct LVM on CentOS

Use a (internal) Docker Registry ASAP

Wrapping up

Microservices are difficult

Docker in production is not so difficult

Embrace DevOps practices

Adopt what you can control