

Kubernetes Serverless with Knative

Burr Sutter (bursutter.com)

<http://bit.ly/knative-tutorial>

Upcoming 3 hour classes/workshops

9 Steps to Awesome with Kubernetes

April 19, 2019

<https://www.safaribooksonline.com/live-training/courses/9-steps-to-awesome-with-kubernetes/0636920257790/>

May 21, 2019

<https://learning.oreilly.com/live-training/courses/9-steps-to-awesome-with-kubernetes/0636920258643/>

Istio on Kubernetes: Enter the Service Mesh

April 12, 2019

<https://www.safaribooksonline.com/live-training/courses/istio-on-kubernetes-enter-the-service-mesh/0636920258681/>

May 22, 2019

<https://learning.oreilly.com/live-training/courses/istio-on-kubernetes-enter-the-service-mesh/0636920258735/>

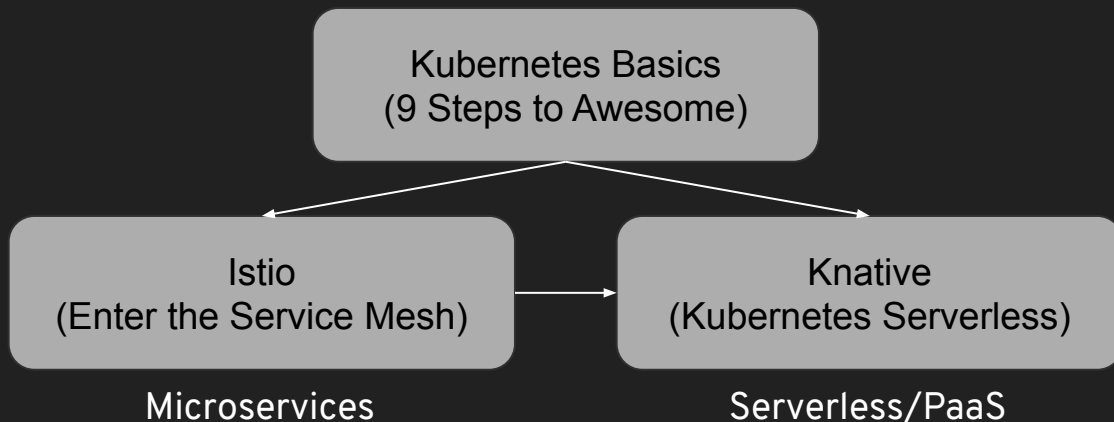
Kubernetes Serverless with Knative

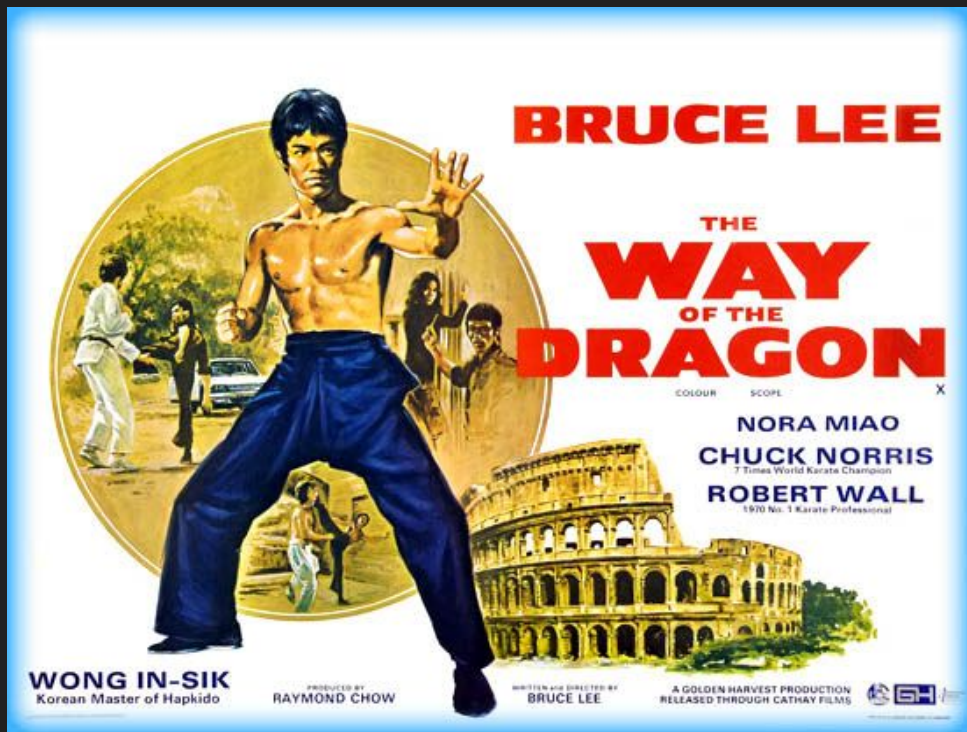
April 17, 2019

<https://www.safaribooksonline.com/live-training/courses/kubernetes-serverless-with-knative/0636920258827/>

May 29, 2019

<https://learning.oreilly.com/live-training/courses/kubernetes-serverless-with-knative/0636920258865/>





<http://gonewiththetwins.com/new/way-dragon-return-dragon-1972/>



<http://www.fanpop.com/clubs/bruce-lee/images/27605335/title/way-dragon-photo>

Exercise Setup

<https://github.com/redhat-developer-demos/knative-tutorial>

Testing/Demo Scripts

<https://github.com/burrsutter/scripts-knative>

<https://github.com/burrsutter/sidebyside>

Knative Tutorial Exercises (bit.ly/knative-tutorial)

- Setup
- Serving: Deploy Knative Service & Revisions
- Configurations & Routes
- Auto-scaling
- Build
- Eventing

Knative's Primary Components

Build

Serving

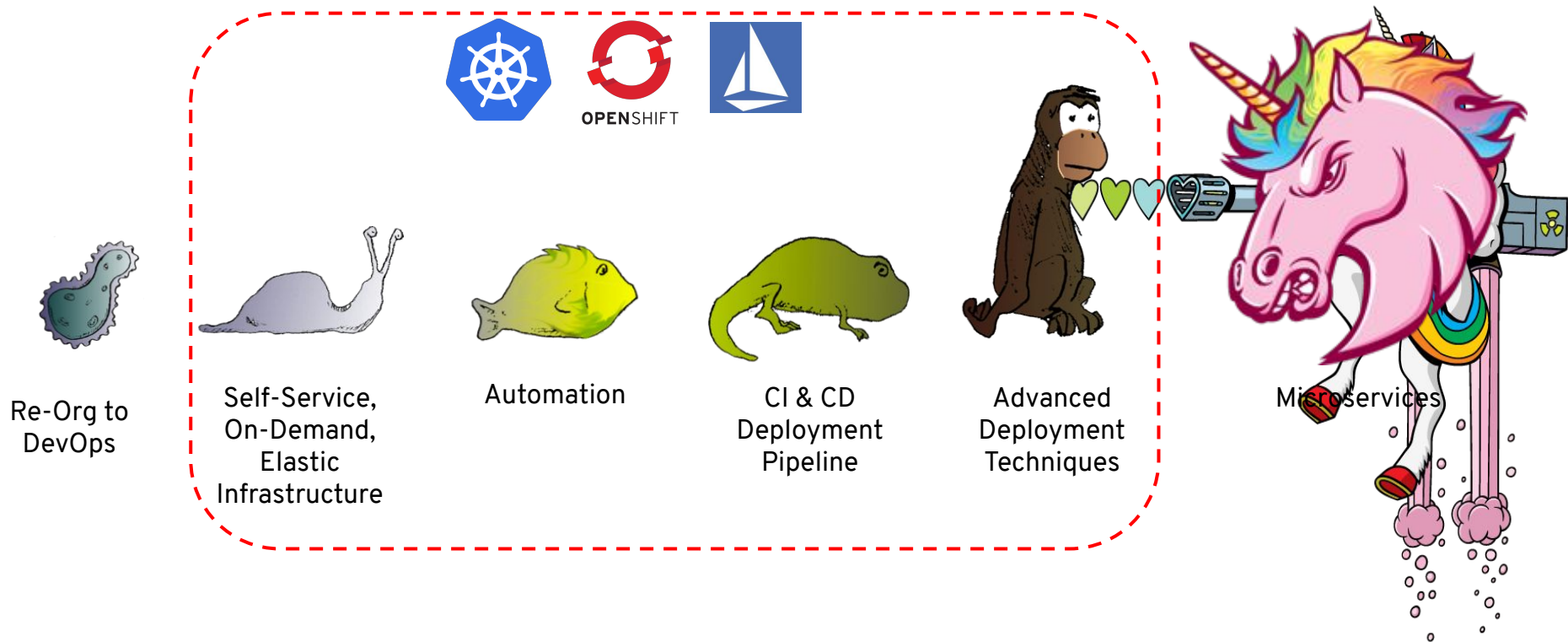
Eventing

Agenda

- What is Serverless
- Serverless vs FaaS
- Serverless via BaaS & SaaS
- FaaS Introduction
- Knative Serving
- Knative Build
- Knative Eventing

Are custom software Apps/APIs a key
strategic advantage for your organization?
OR
Do you regard IT as a cost center that must
be better streamlined?

Your Journey to Awesomeness

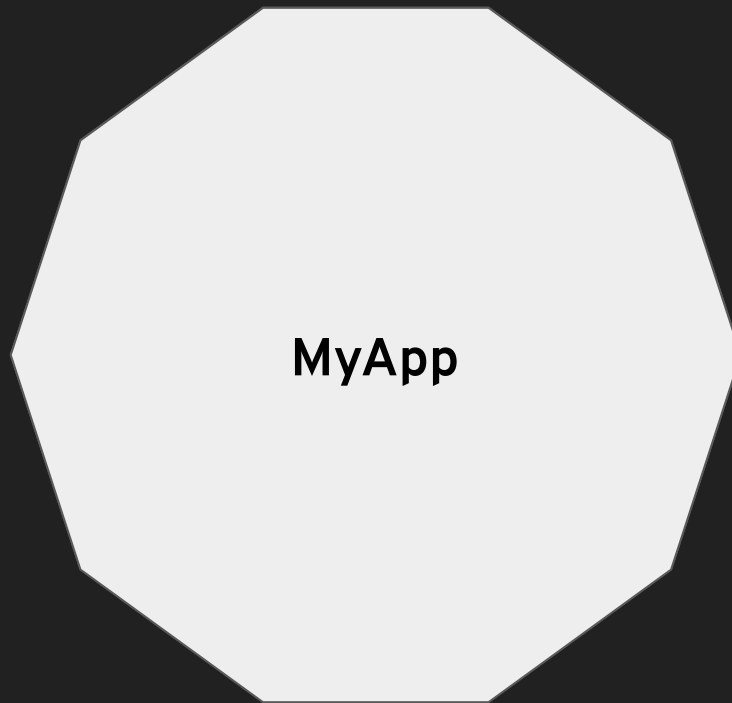




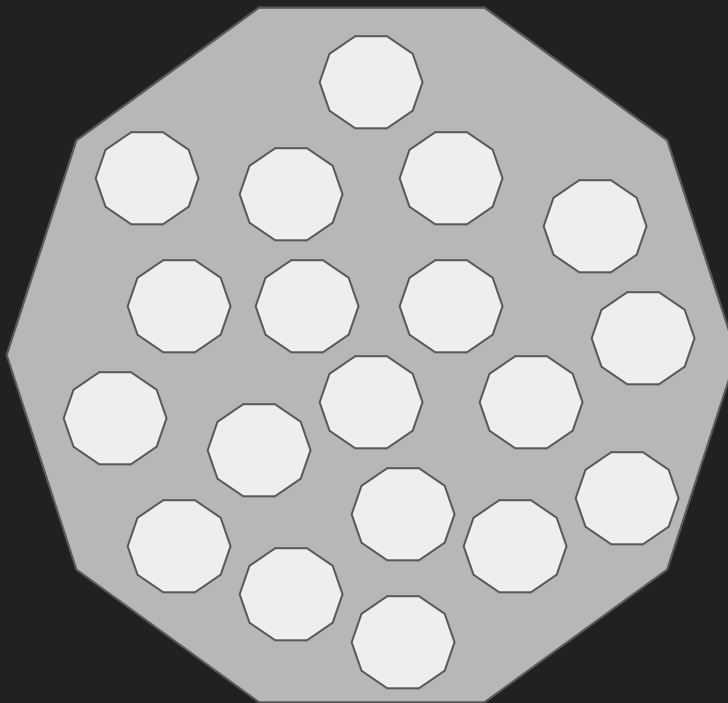
We cannot solve our problems
with the same thinking we used
when we created them.

Albert Einstein
(Theoretical Physicist)

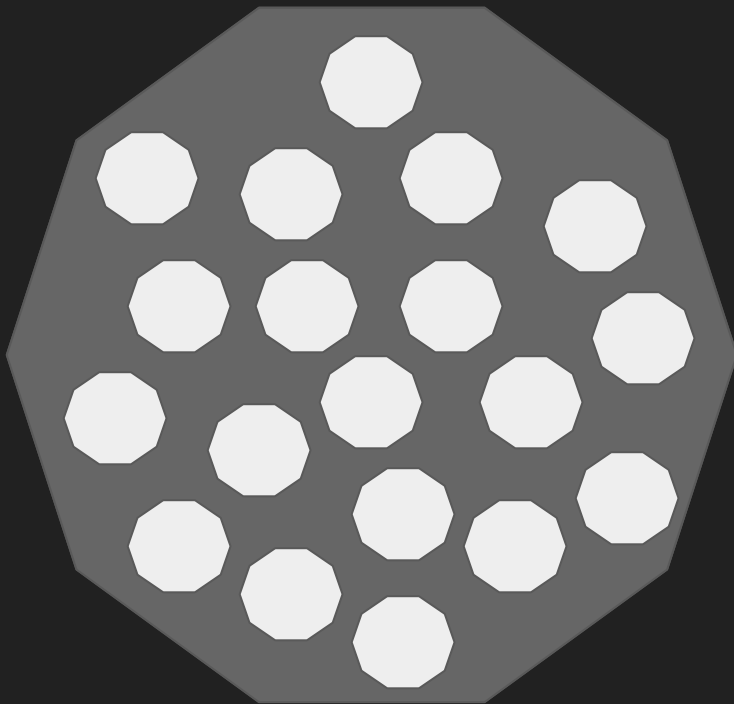
Monolith



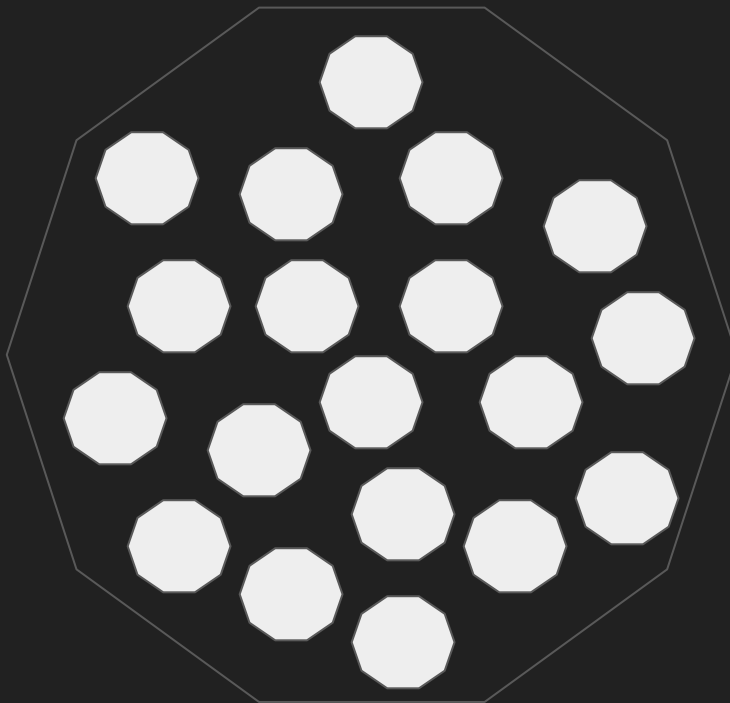
The Application



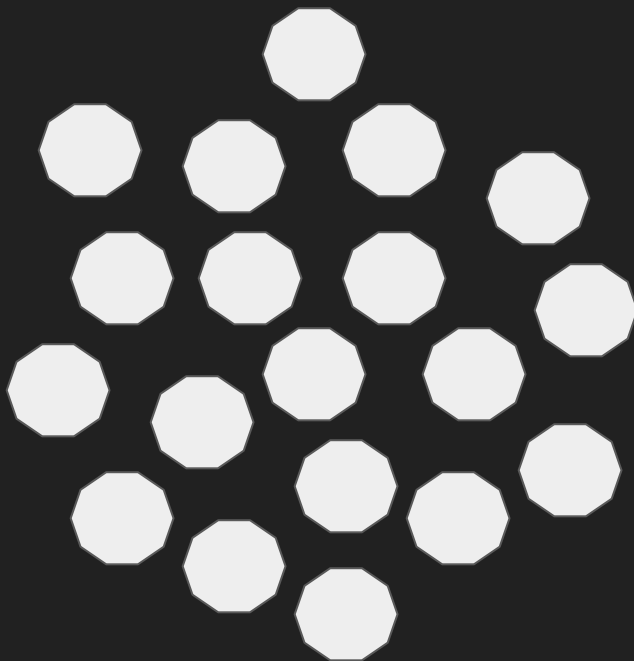
Modules



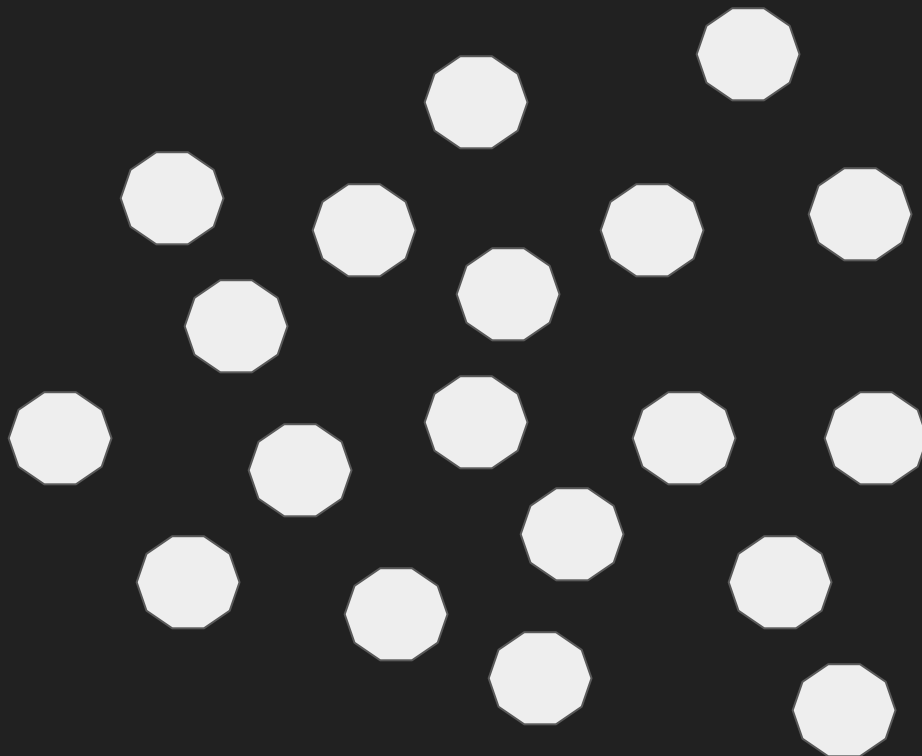
Microservices



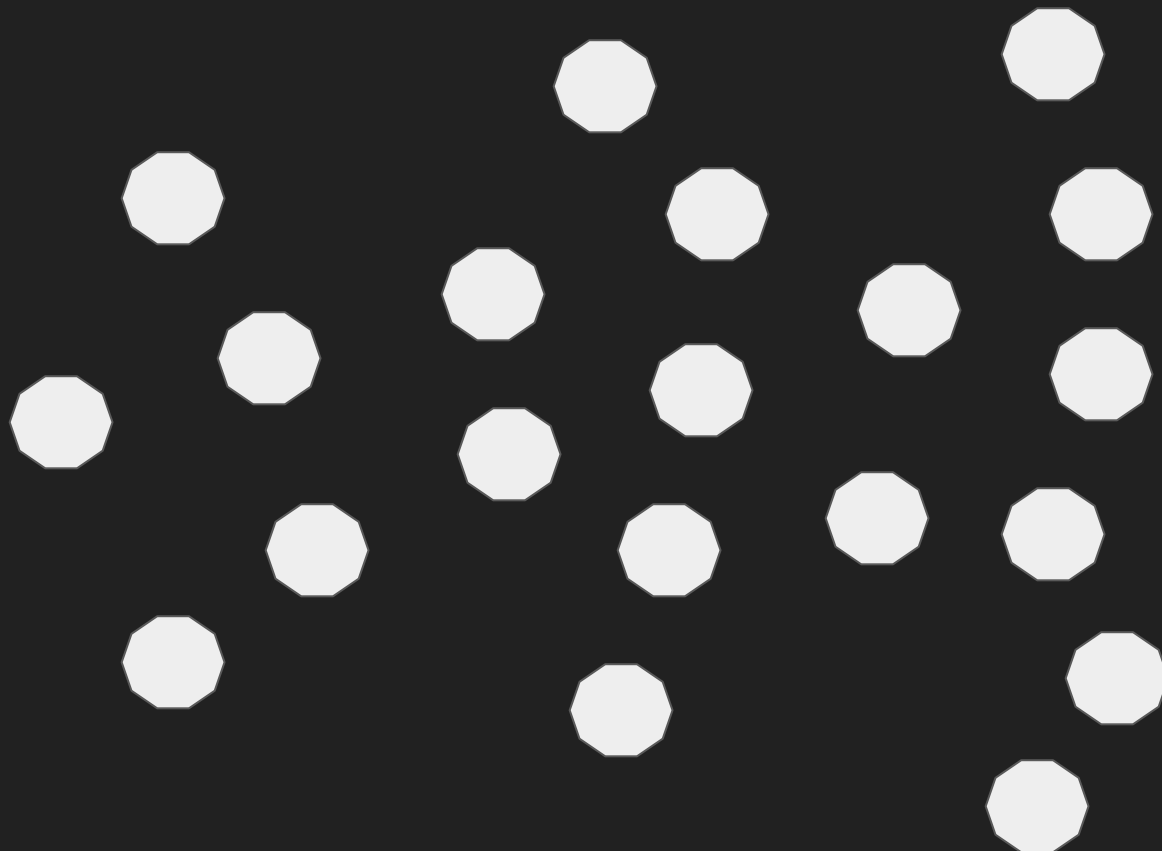
Microservices



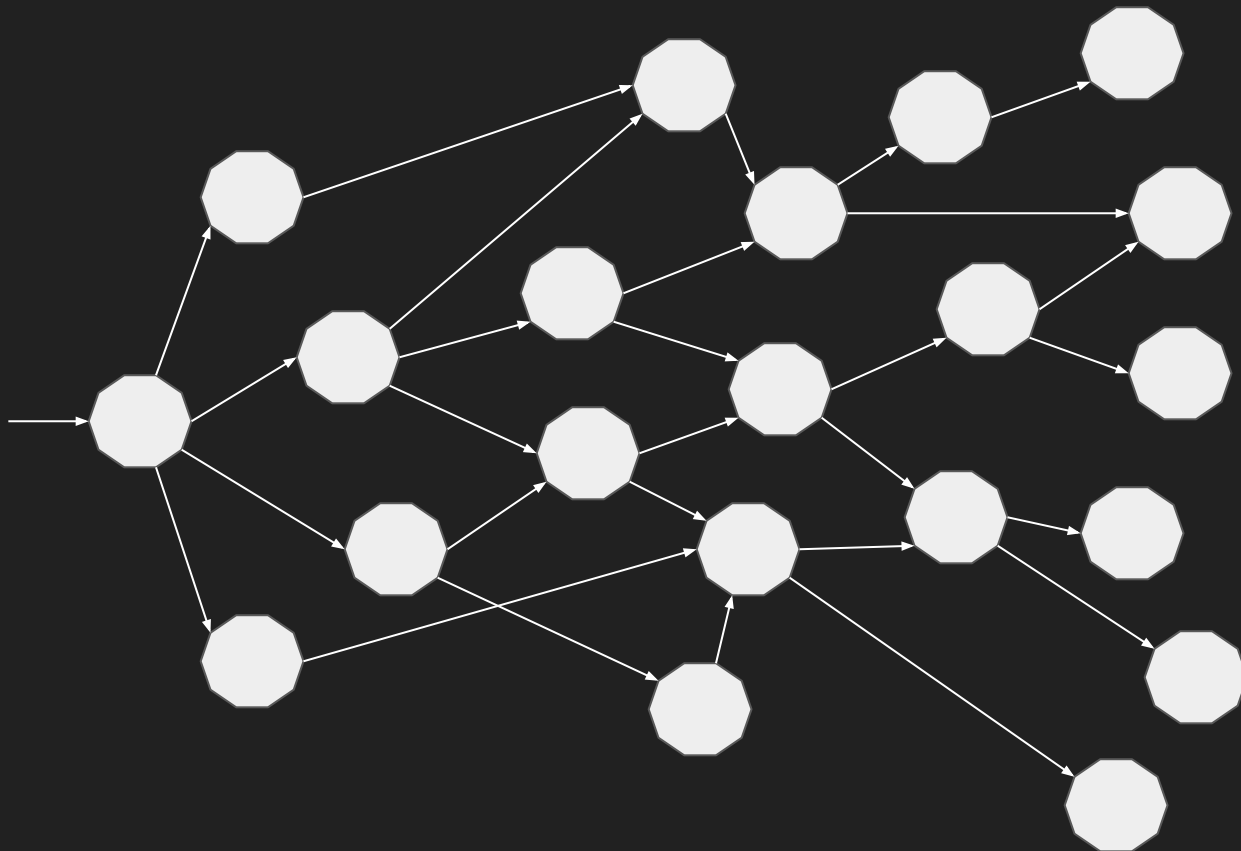
Microservices



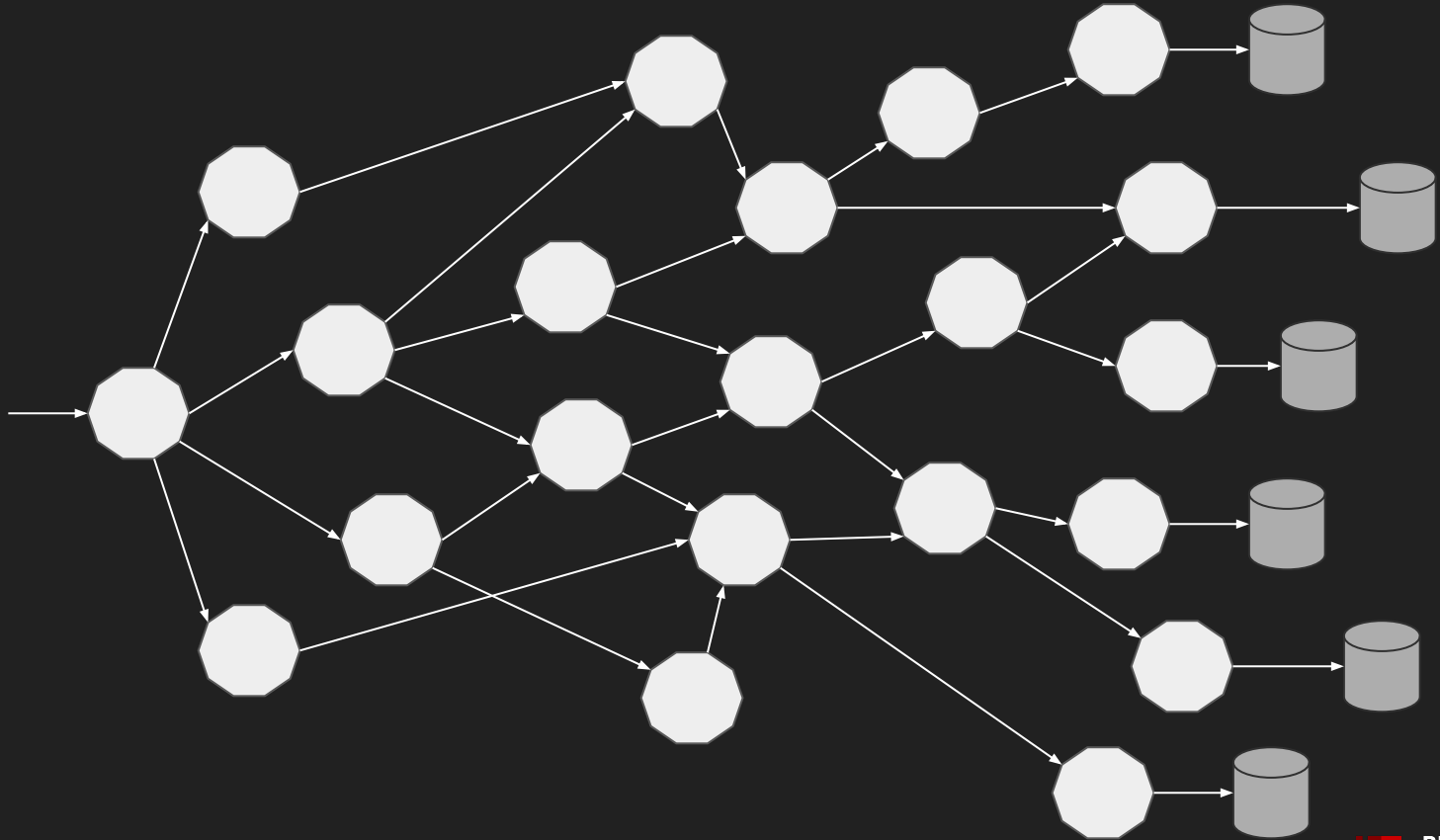
Microservices



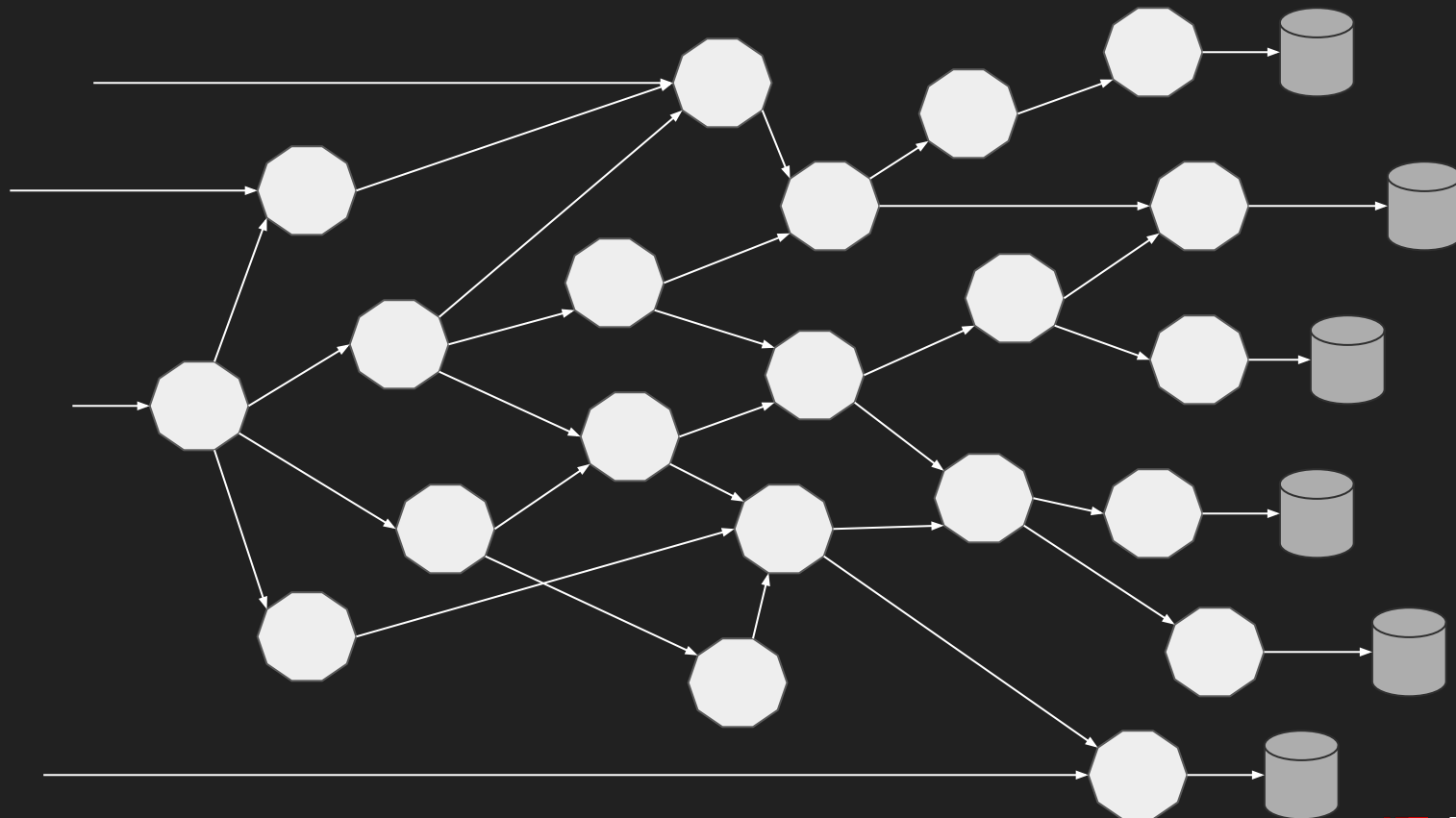
Network of Services



Microservices own their Data



Multiple Points of Entry

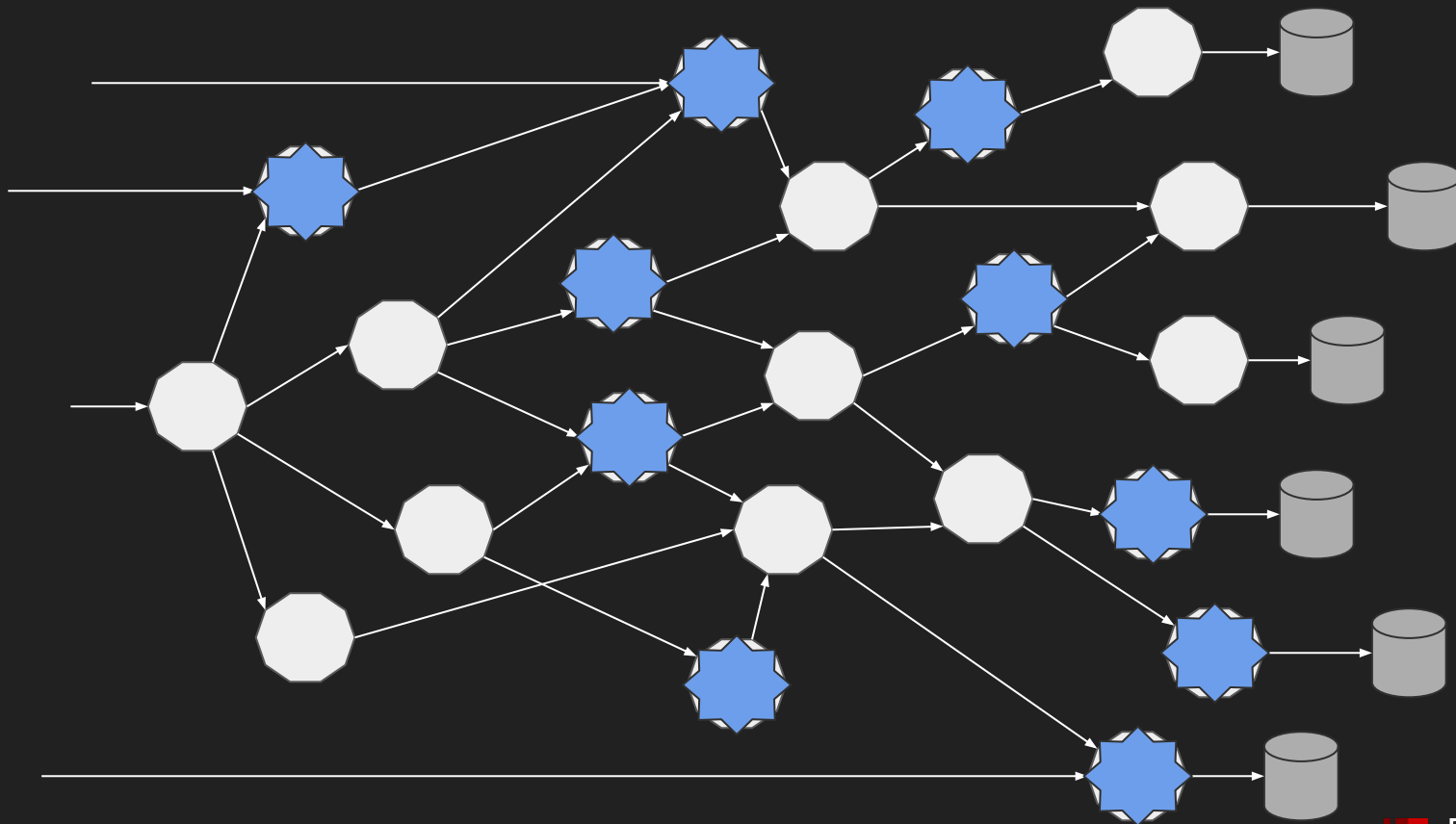




“Change is the essential
process of all of existence.”

—SPOCK

Let there be Functions?



Serverless

<https://www2.eecs.berkeley.edu/Pubs/TechRpts/2019/EECS-2019-3.pdf>

Cloud Native Computing Foundation

“Serverless computing refers to the concept of building and running applications that do not require server management. It describes a finer-grained deployment model where applications, bundled as one or more functions, are uploaded to a platform and then executed, scaled, and billed in response to the exact demand needed at the moment.”

<https://www.cncf.io/blog/2018/02/14/cncf-takes-first-step-towards-serverless-computing/>

Serverless vs FaaS

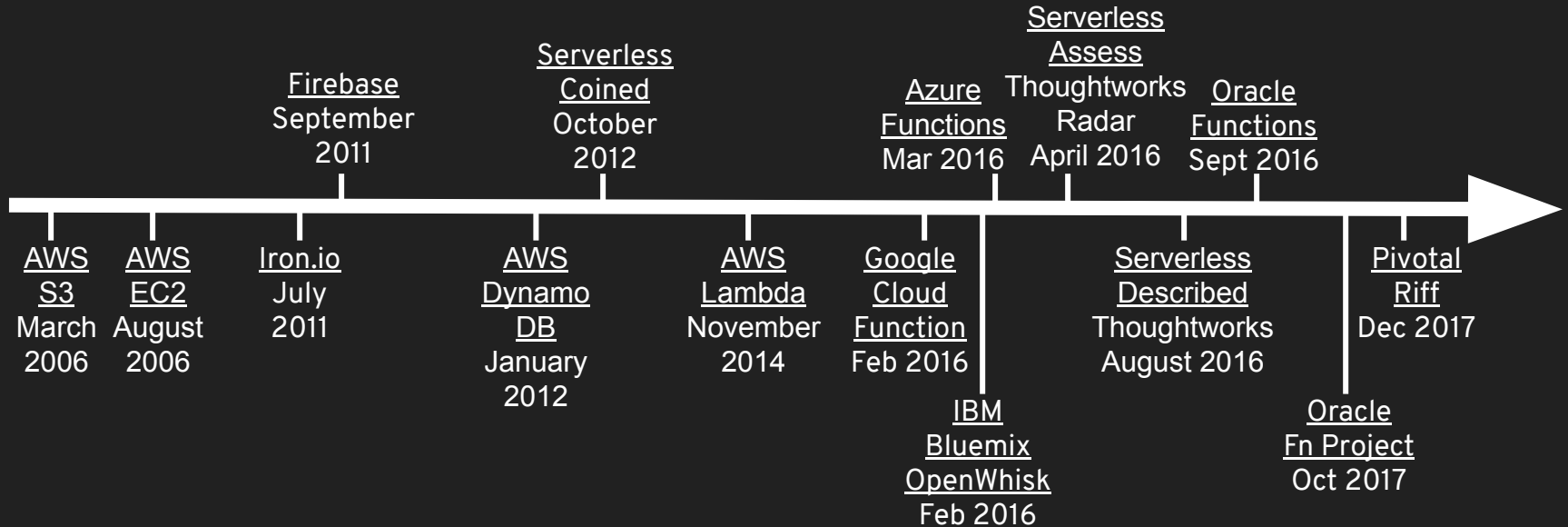
'...are application designs that incorporate third-party “Backend as a Service” (BaaS) services, and/or that include custom code run in managed, ephemeral containers on a “Functions as a Service” (FaaS) platform'

<https://martinfowler.com/articles/serverless.html>

'The survey defined FaaS as typically providing event-driven computing where developers run and manage application code with functions that are triggered by events'

<https://thenewstack.io/add-it-up-serverless-faas/>

Short History of Serverless



The first question is “is there a suitable service I can consume?” before “is there something I can buy and set up using a cloud provider?”.

[Thoughtworks Nov 29 2017](#)

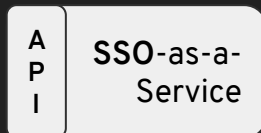
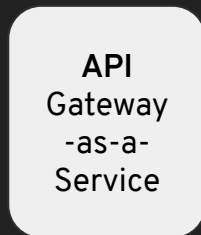
It is Serverless
because of BaaS/SaaS
(managed by another party services).

It is all about the Services

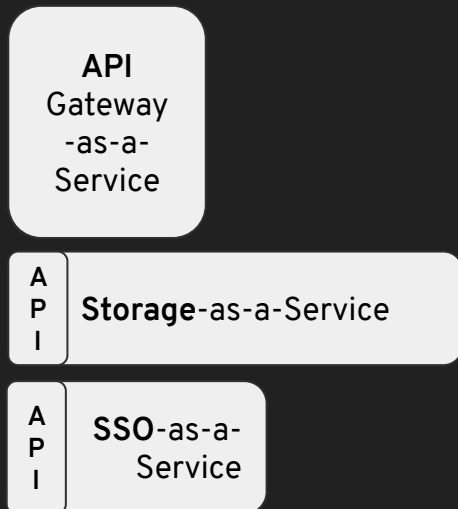
HTTP Input/Output Service

API
Gateway
-as-a-
Service

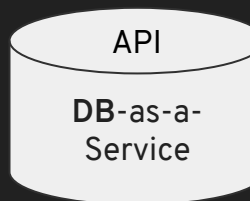
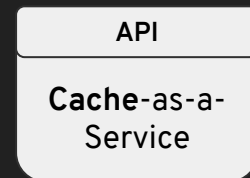
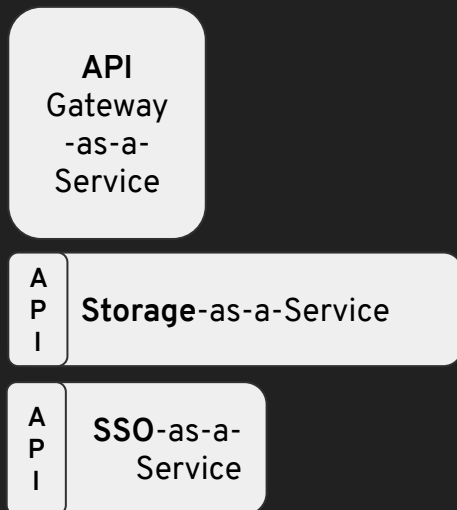
Authentication Service



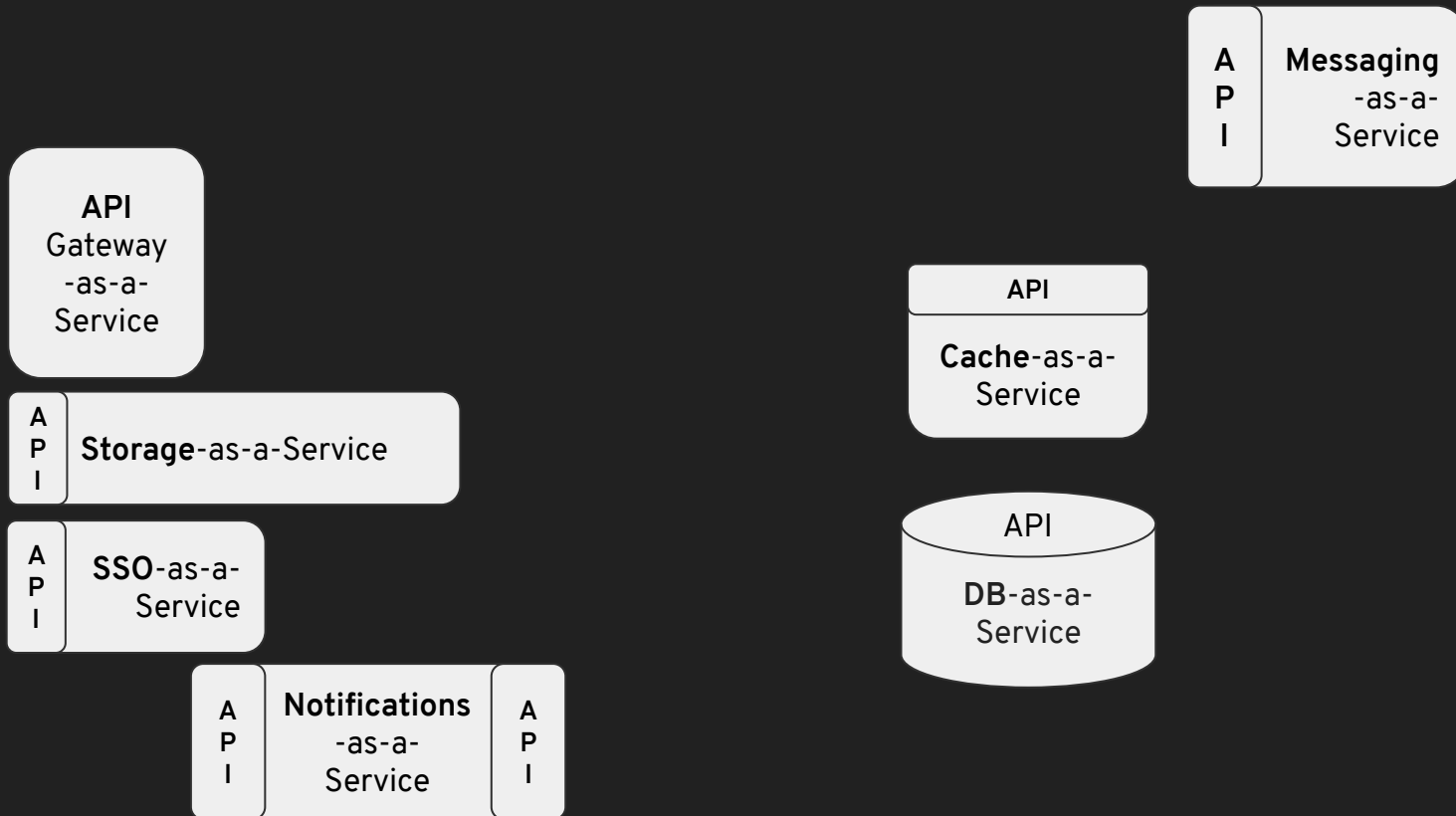
File Storage Service



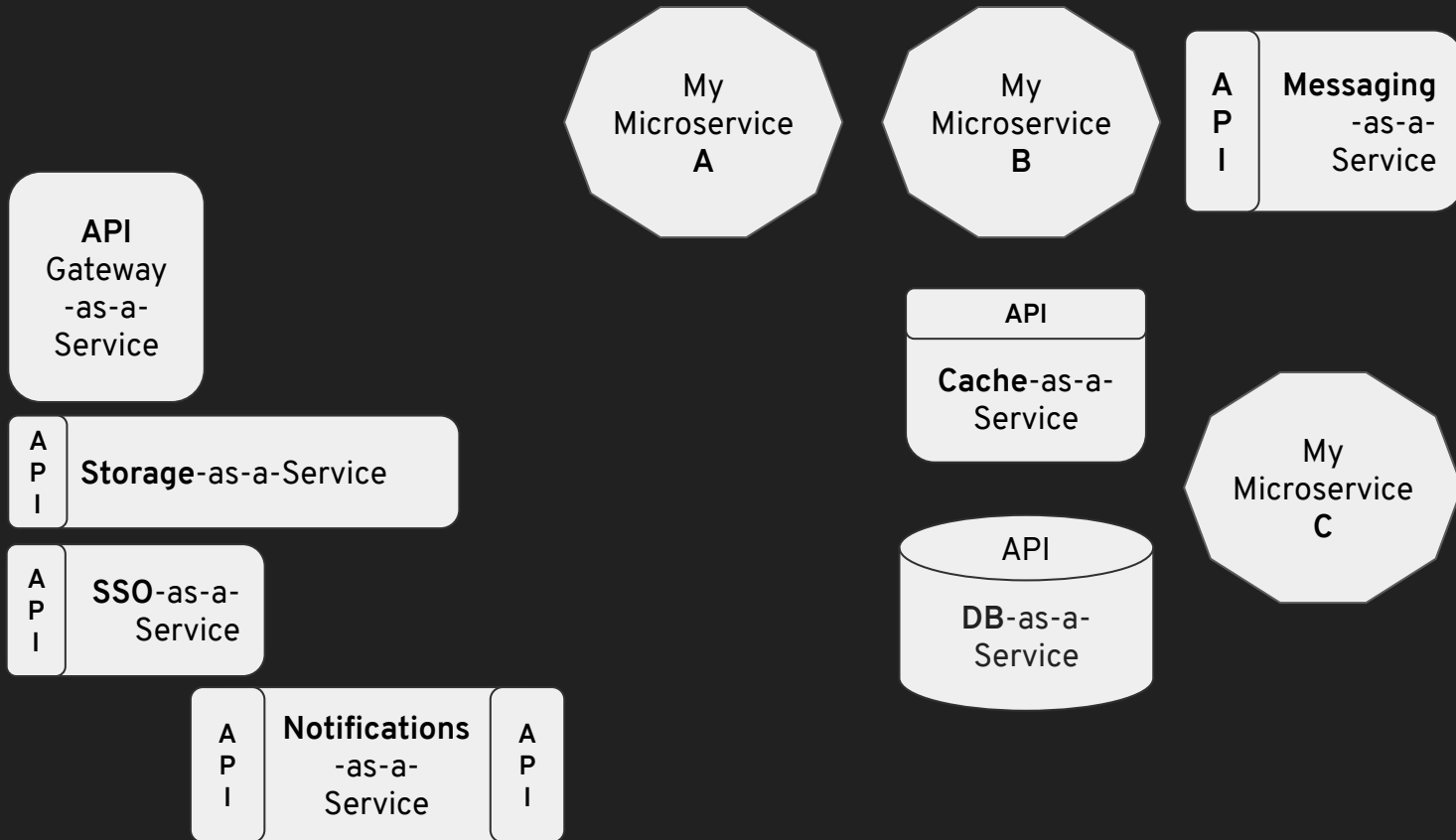
Data Services



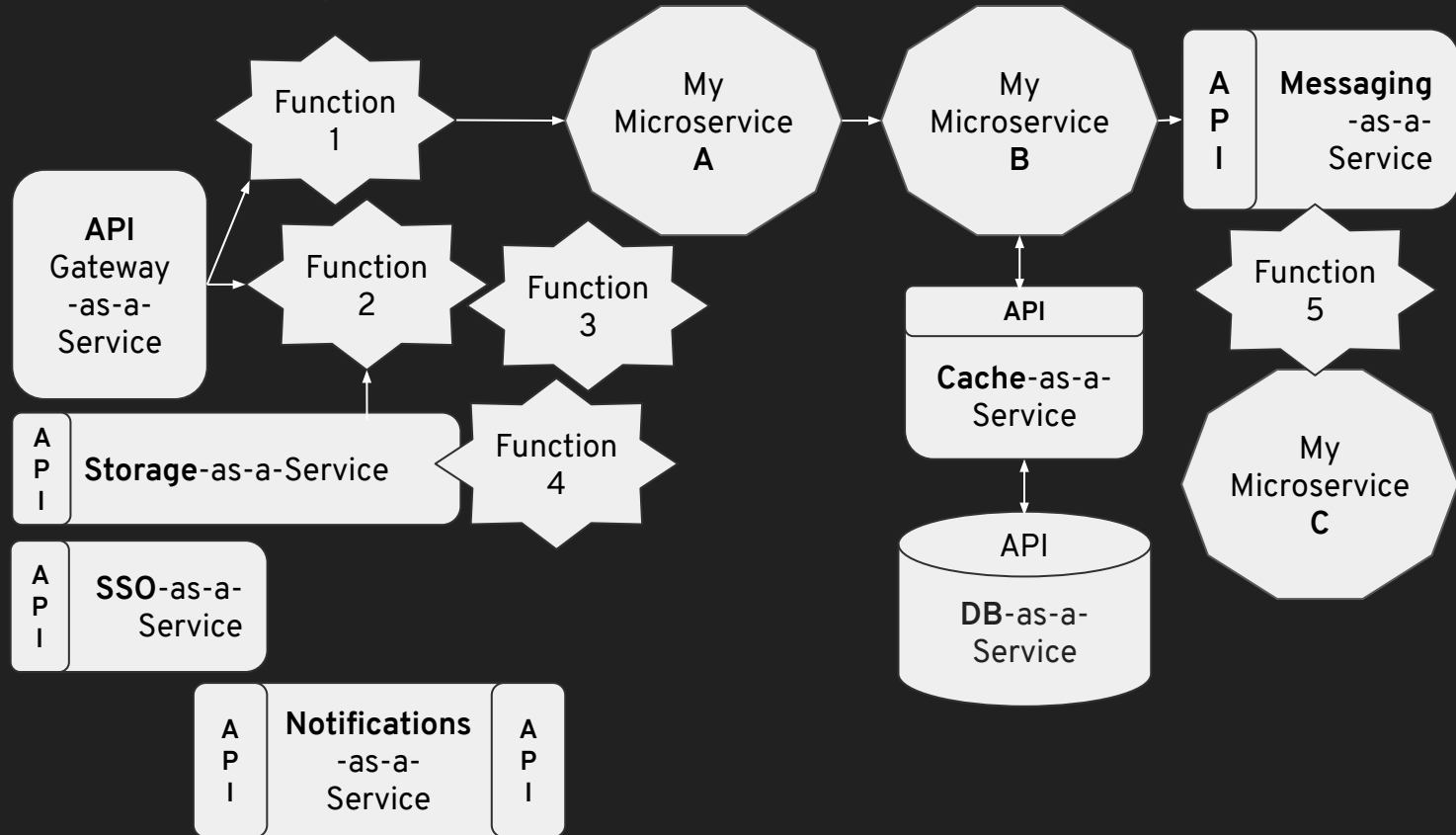
Connectivity Services



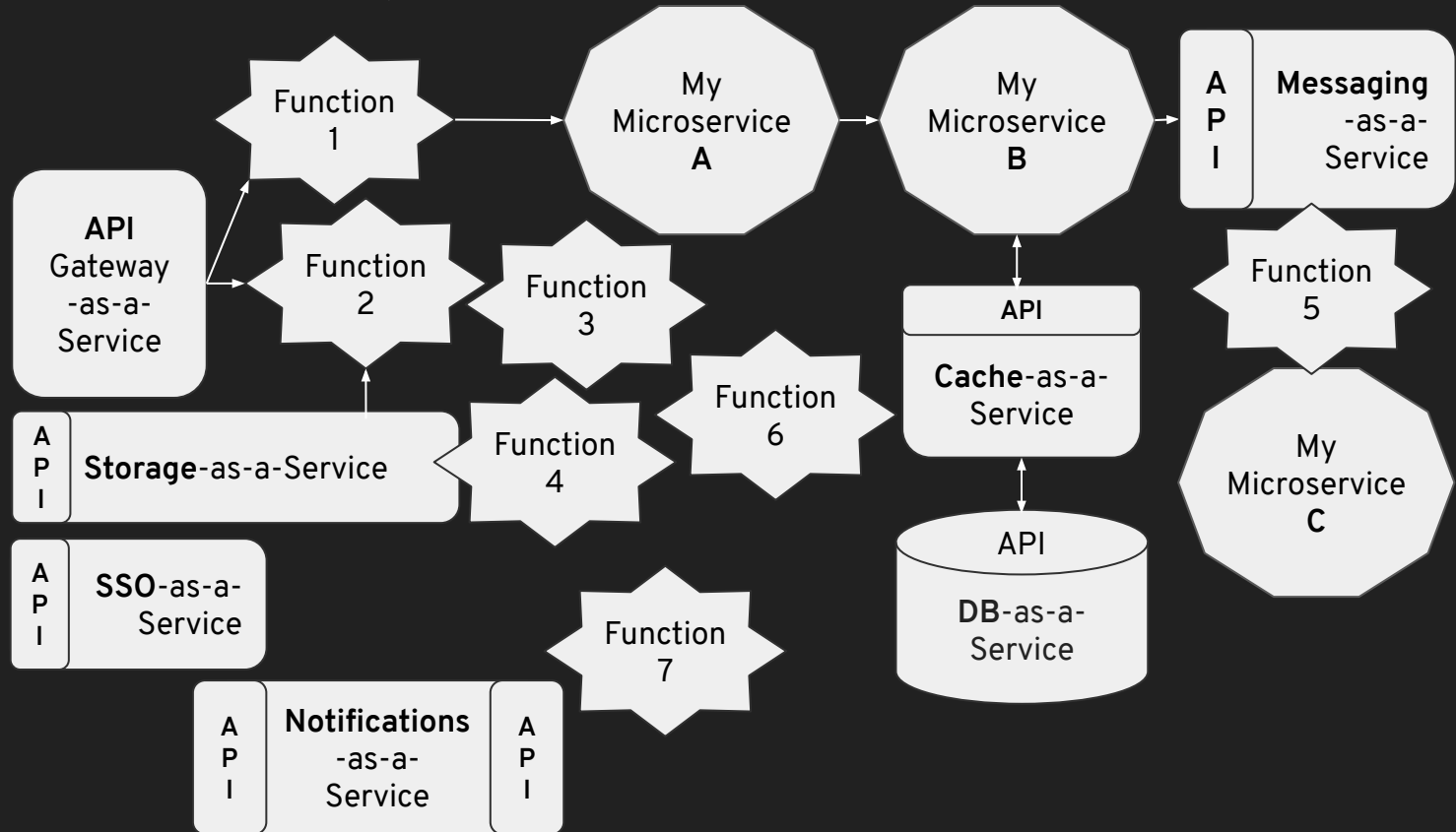
Your Containerized Services



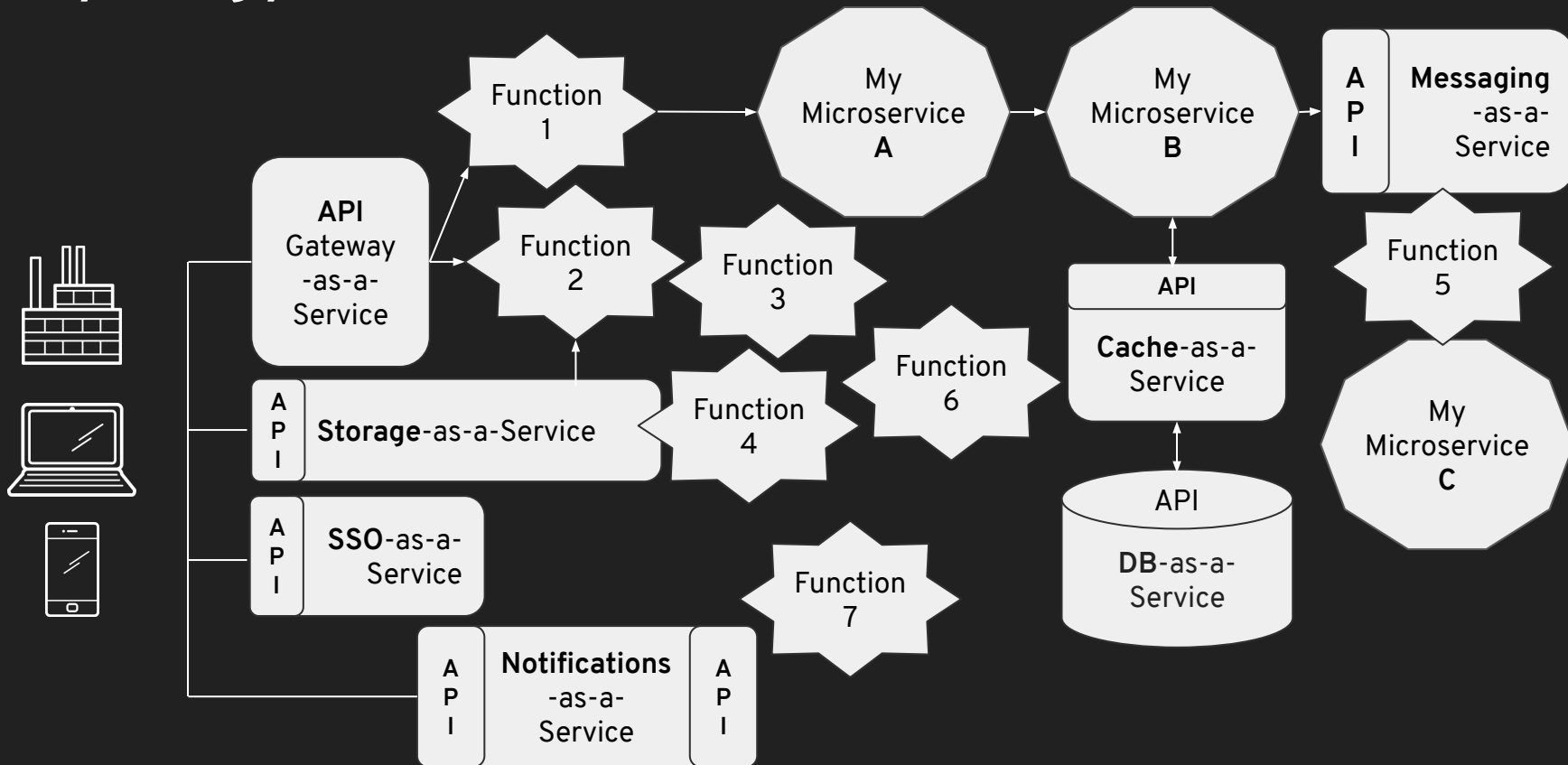
Event-Driven Input



Event-Driven Output



Synergy



Microservices

Serverless Functions



Your Control
Long-Lived Processes
Known Programming Model
Often Sync Request-Response

Mature:
IDE Integration
Debuggers
Tracers
Monitoring
CI/CD

Cloud Control
Short-Lived Processes
New Programming Model
Event-Driven Async

Immature:
?

FaaS

FaaS Kubernetes Players



Kubernetes/OpenShift Review

```
mvn package  
docker build  
kubectl apply -f deploy.yml  
kubectl apply -f service.yml
```

Knative



<https://github.com/knative>

Knative Announced July 24



FaaS Kubernetes Players



APACHE
OpenWhisk™



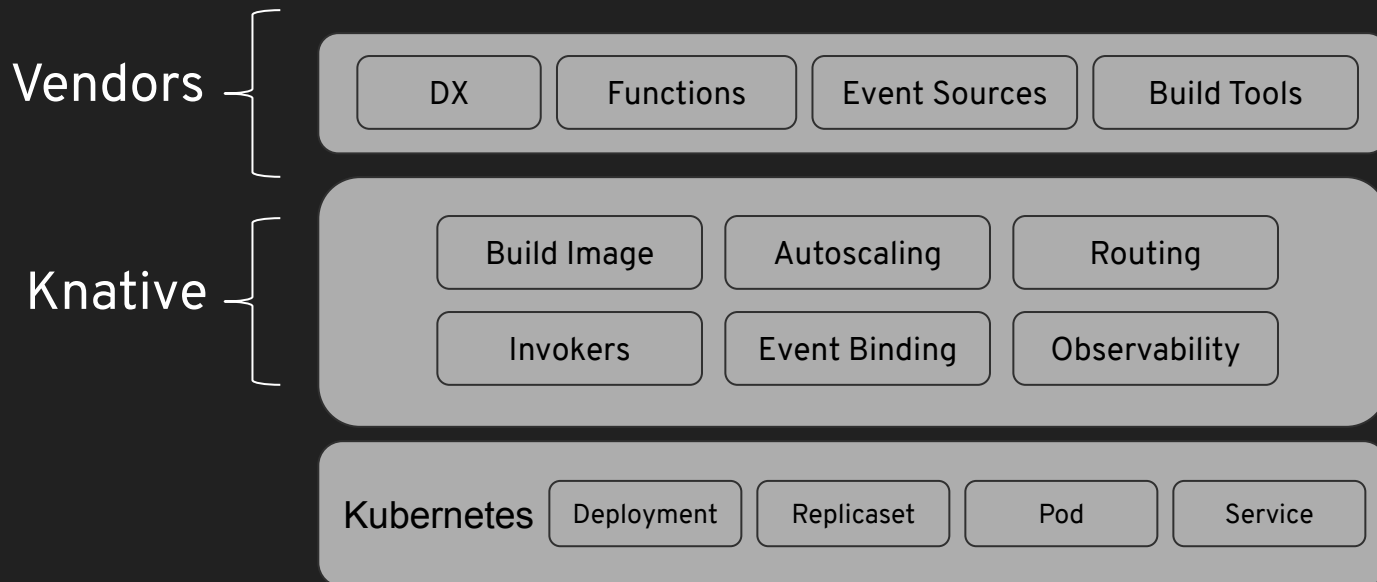
Why?



OPENFAAS

nuclio

Primitives



What is Knative?

"Kubernetes-based platform to build, deploy, and manage modern **serverless** workloads."

"Essential **base primitives** for all"

"Knative provides a set of **middleware components** that are essential to build modern, source-centric, and **container-based applications** that can run anywhere: on premises, in the cloud, or even in a third-party data center"

Interesting Capabilities

- Scale-to-zero: No pod == no memory
- Scale-from-zero: Traffic spike starts N pods
- Configurations & Revisions - built-in Blue/Green
- In-Cluster Image Building
- Traffic splitting
- Eventing System



Exercises

bit.ly/knative-tutorial

<https://github.com/burrsutter/scripts-knative>

Knative Serving

Knative Serving

```
kubectl get crd | grep serving
```

```
configurations.serving.knative.dev
```

```
revisions.serving.knative.dev
```

```
routes.serving.knative.dev
```

```
services.serving.knative.dev
```

Knative Serving Autoscaler

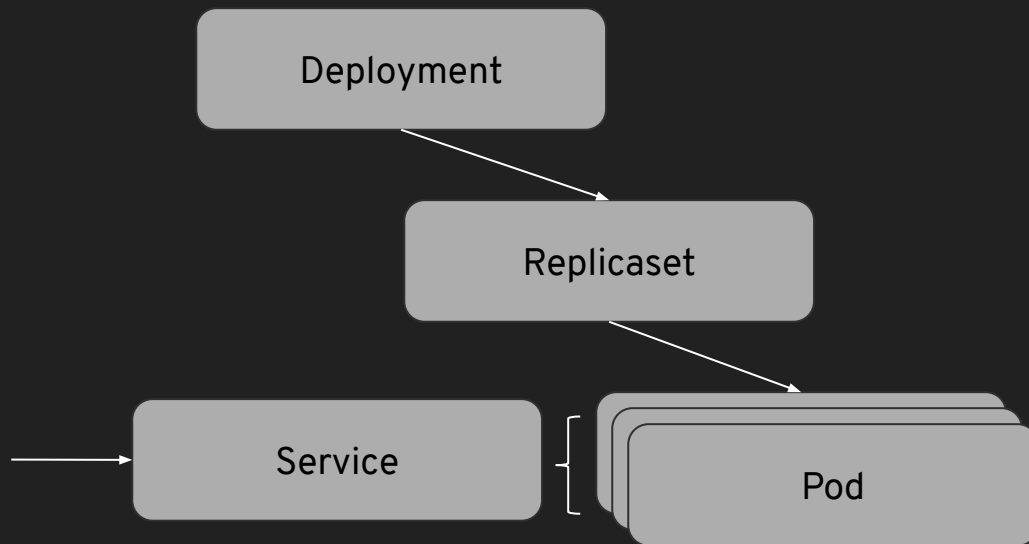
```
kubectl -n knative-serving edit  
configmap config-autoscaler
```

```
container-concurrency-target-default: "1"
```

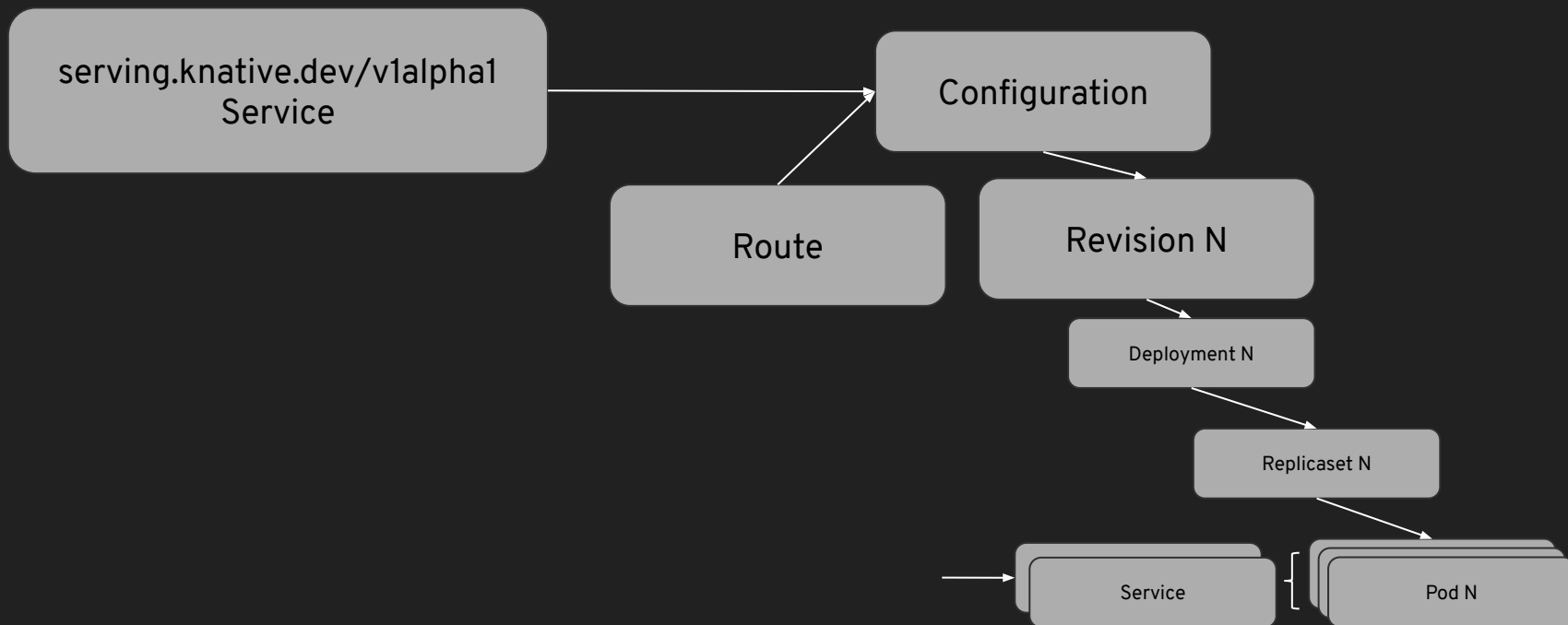
```
scale-to-zero-grace-period: 30s
```

```
stable-window: 30s
```

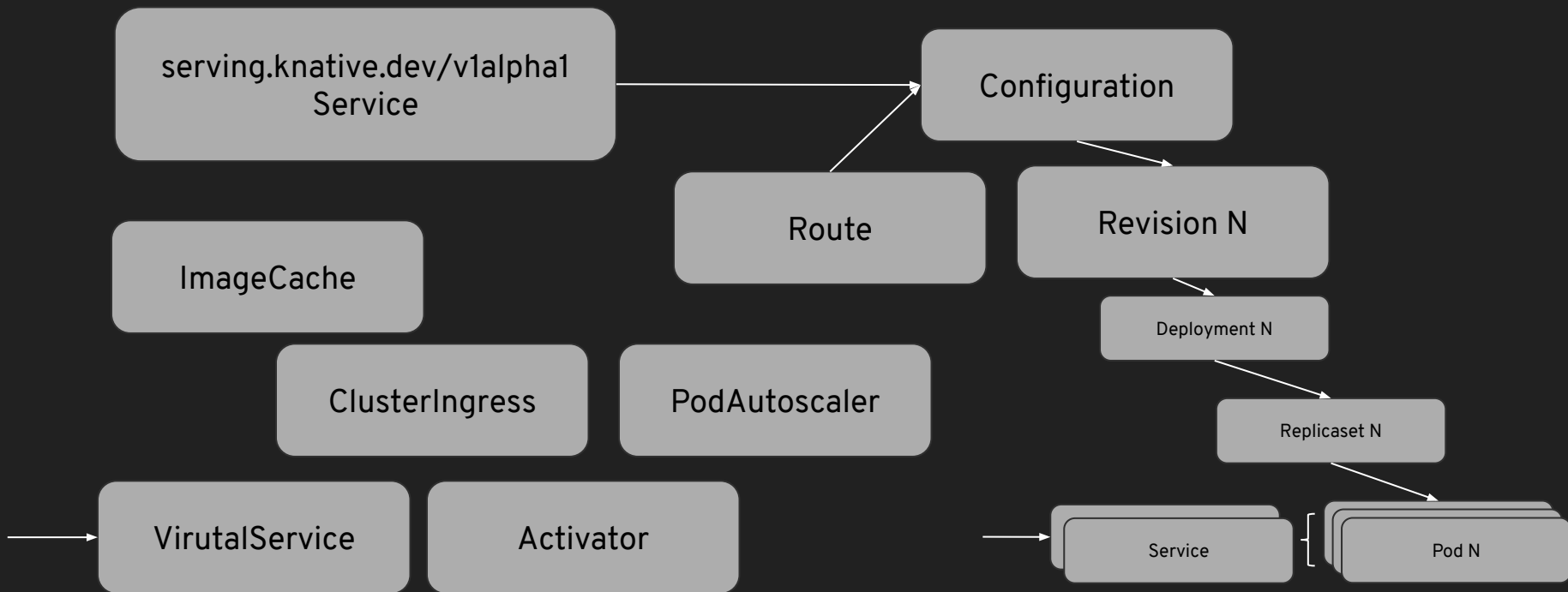
kubectl apply -f Deployment.yaml



kubectl apply -f ksvc.yaml



kubectl apply -f ksvc.yaml



Knative Build

Knative Build

```
kubectl get crd | grep build
```

```
builds.build.knative.dev
```

```
buildtemplates.build.knative.dev
```

```
clusterbuildtemplates.build.knative.dev
```

Knative Build Resources

1. Secret

User ID/Password for remote registry

2. ServiceAccount

Associated with the Secret

3. PVC: Maven

Caches Maven downloads

4. PVC: Image Cache

Caches Linux image downloads

5. Build

The Steps

6. Service

Knative Serving Service



TEKTON

Knative Eventing

Knative Eventing

```
kubectl get crd | grep eventing
```

```
channels.eventing.knative.dev
```

```
subscriptions.eventing.knative.dev
```

```
cronjobsources.sources.eventing.knative.dev
```

```
githubsources.sources.eventing.knative.dev
```

```
kuberneteseventsources.sources.eventing.knative.dev
```

```
containersources.sources.eventing.knative.dev
```

Resources

<https://developers.redhat.com/blog/2019/04/09/from-zero-to-quarkus-and-knative-the-easy-way/>

<https://blog.openshift.com/knative-building-your-serverless-service/>

<https://blog.openshift.com/knative-serving-your-serverless-services/>

<https://blog.openshift.com/knative-configurations-routes-and-revisions/>

O'REILLY®



Compliments of
RED HAT
DEVELOPERS

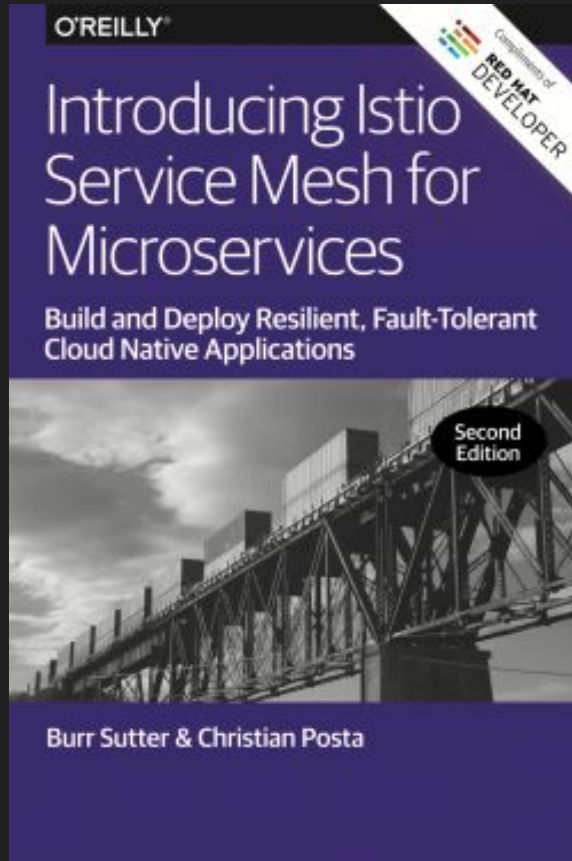
Migrating to Microservice Databases

From Relational Monolith
to Distributed Data



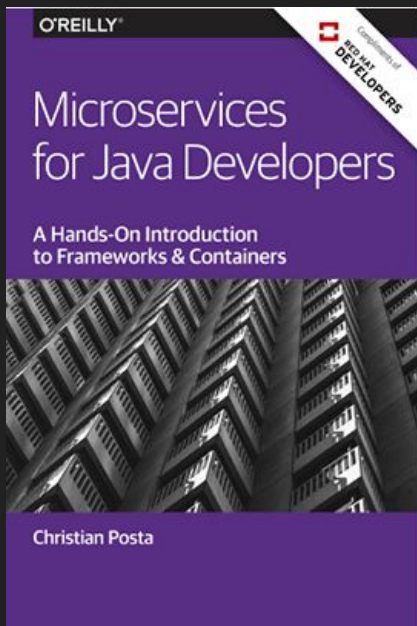
Edson Yanaga

bit.ly/mono2microdb



bit.ly/istiobook

bit.ly/javamicroservicesbook



Free eBooks from developers.redhat.com

Microservices Introductory Materials

Demo: bit.ly/msa-instructions

Slides: bit.ly/microservicesdeepdive

Video Training: bit.ly/microservicesvideo

[Kubernetes for Java Developers](#)

[9 Steps to Awesome with Kubernetes](#)

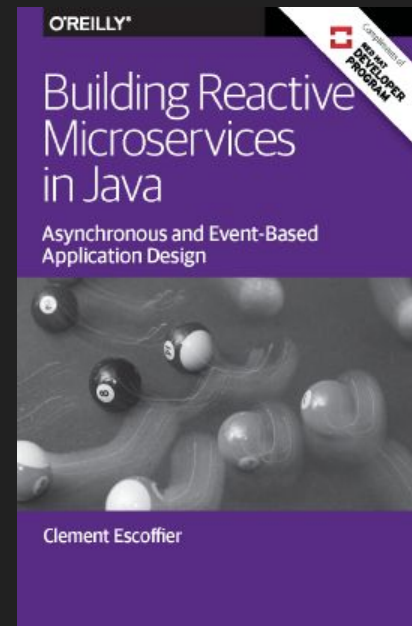
Advanced Materials

bit.ly/istio-tutorial

learn.openshift.com/servicemesh

bit.ly/knative-tutorial

bit.ly/reactivemicroservicesbook



bit.ly/serverlesskube

@bursutter - bit.ly/serverlesskube