

Azure Kubernetes Service con .NET 5

Alessandro Melchiori

Founder & Software developer @ CodicePlastico
alessandro@codiceplastico.com | @amelchiori

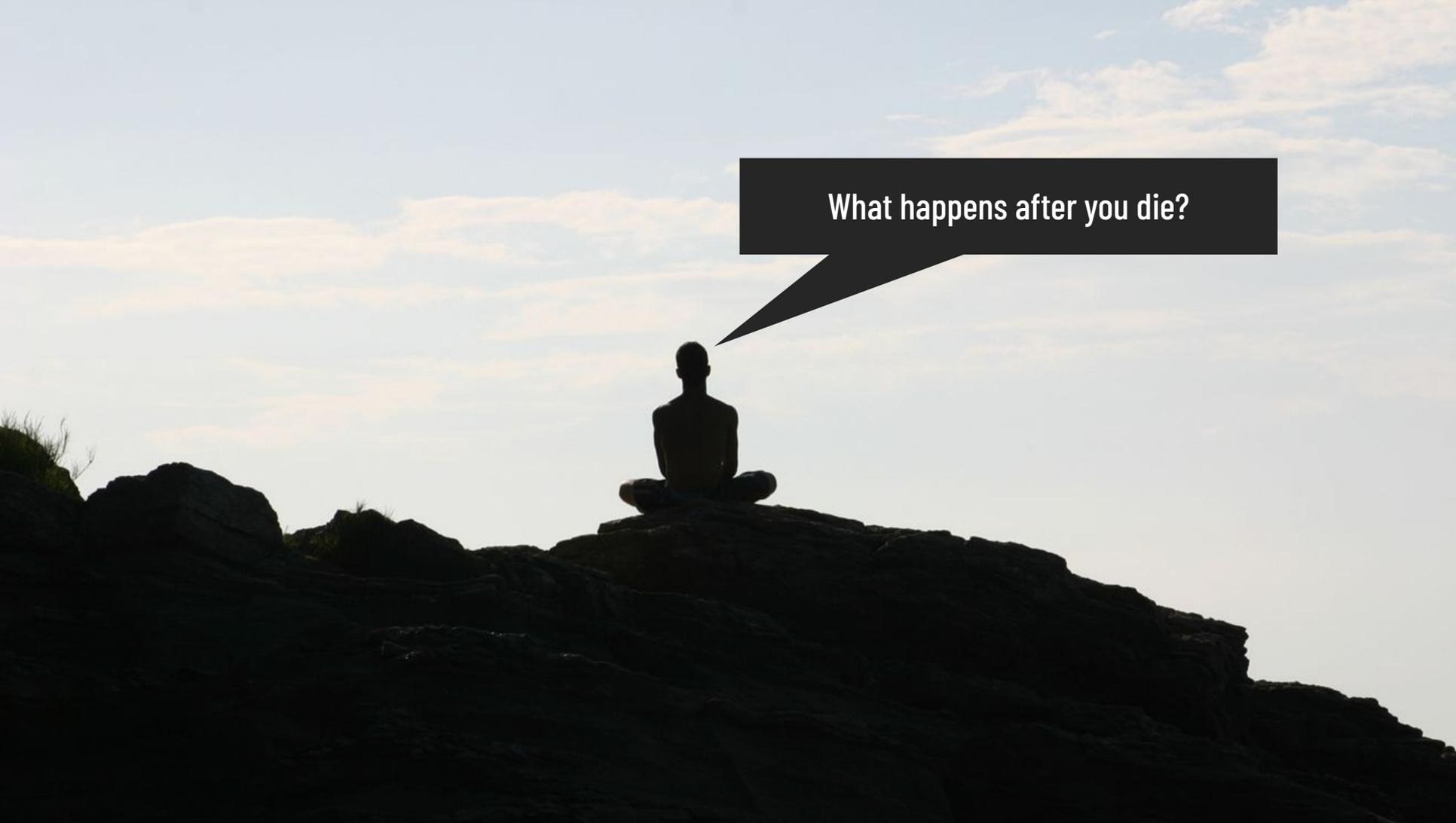
.NET Conference
Italia 2020

.NET



aspitalia.com



A silhouette of a person sitting in a meditative pose on a rocky cliff. The person is centered in the lower half of the frame, facing away from the viewer. The background is a vast, cloudy sky with soft, diffused light. A dark, rectangular speech bubble with a tail pointing to the person's head is positioned in the upper right quadrant. Inside the speech bubble, the text "What happens after you die?" is written in a white, sans-serif font.

What happens after you die?

A silhouette of a person sitting in a meditative pose on a rocky peak. A speech bubble points to the person, containing the text "What is life?". The background is a sky with light clouds.

What is life?

A silhouette of a person sitting in a meditative pose on a dark, rocky outcrop. The person is centered in the lower half of the frame. Above them, a red speech bubble points towards their head. The background is a vast, open sky with scattered, light-colored clouds, suggesting a dawn or dusk setting. The overall mood is contemplative and serene.

What is a microservice?



Adam Sándor
@adamsand0r

Follow

Microservices are units of (independent!) deployment. If you need to deploy several of them together, then you're doing something wrong. Take a good look at your architecture and see if you can make them truly independent. If not, they probably should be in one service.

5:25 PM - 3 Oct 2018

65 Retweets 157 Likes



9

65

157



9

62

121



62 Retweets 121 Likes



5:32 PM - 3 Oct 2018



Kevin Sigmund
@ksigmund

Follow

Replying to @adamsand0r

Microservices are about logical, *not* physical boundaries. We are running multiple services inside the same process, but because they are logically independent, if the need arises, any/all of them could be deployed physically independent of the others.

2:59 AM - 5 Oct 2018

2 Likes



1



2



1



5



Microservice architecture

THE "GOOD"

- An application is sum of its components
- Better fault isolation
- Components can be spread across multiple servers



Microservice architecture

THE "GOOD"

- An application is sum of its components
- Better fault isolation
- Components can be spread across multiple servers

THE "BAD"

- Many components, many moving parts
- Difficult to manage inter-communication
- Manual management can be difficult



Microservice architecture

THE "GOOD"

- An application is sum of its components
- Better fault isolation
- Components can be spread across multiple servers

THE "BAD"

- Many components, many moving parts
- Difficult to manage inter-communication
- **Manual management can be difficult**





Kubernetes 101





Kubernetes

Greek for "Helmsman" < the person who steers a ship





~~Kubernetes~~

Greek for "Helmsman" < the person who steers a ship





K8s

Greek for "Helmsman" < the person who steers a ship

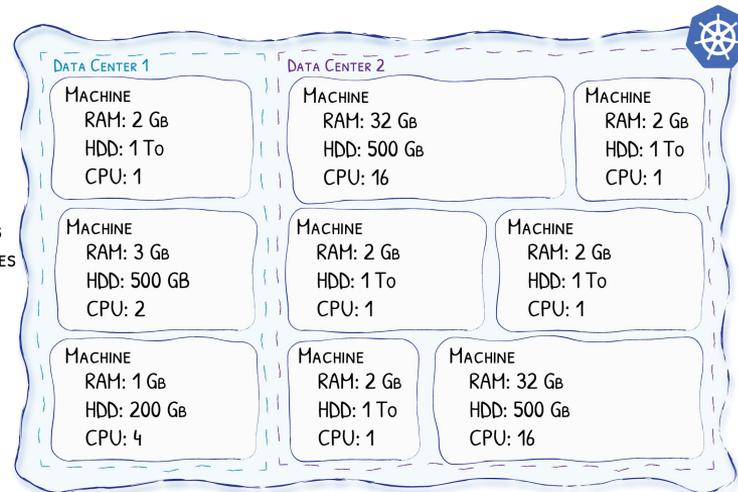


What's Kubernetes (k8s)

Kubernetes is a **cluster technology**.

It means that you will see a cluster of computers as one entity. You will not deploy an application on a specific computer, but *somewhere* in the cluster

A CLUSTER
IS A GROUP
OF MACHINES
THAT BEHAVES
LIKE ONE.



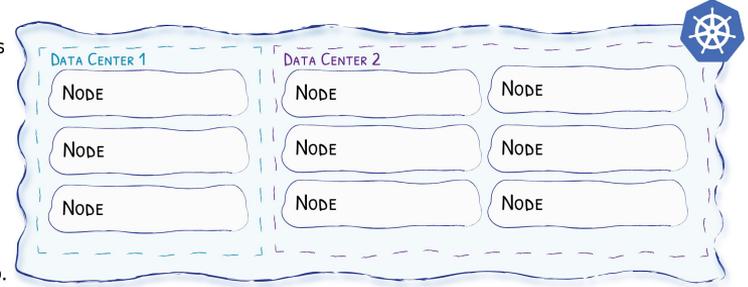
K8s 101 - Nodes

Each computer in the cluster is called a **node**.

Eventually, the nodes will host your applications.
The nodes can be spread throughout the world in different data centers

NODES ARE
THE MACHINES
IN YOUR
CLUSTER.

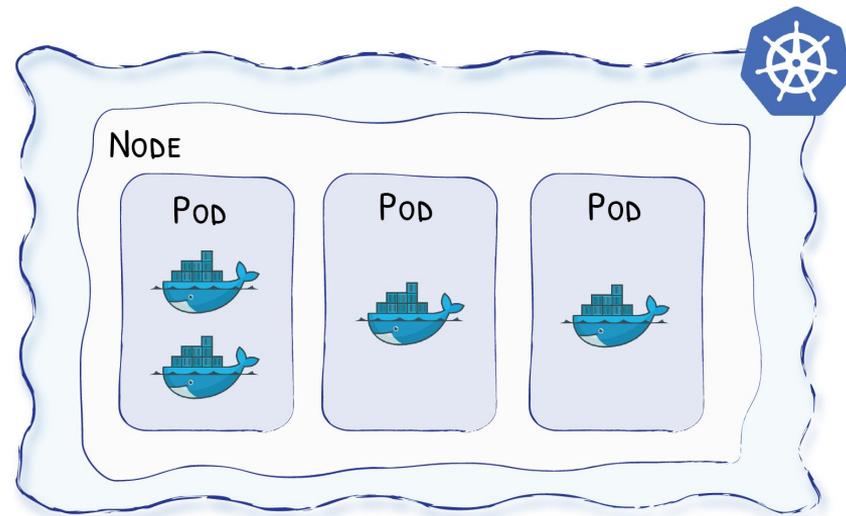
THEY ARE
AVAILABLE
RESOURCES
THAT CAN
COME AND GO.



K8s 101 - Pods

Pods are the smallest unit you will eventually deploy to the cluster.

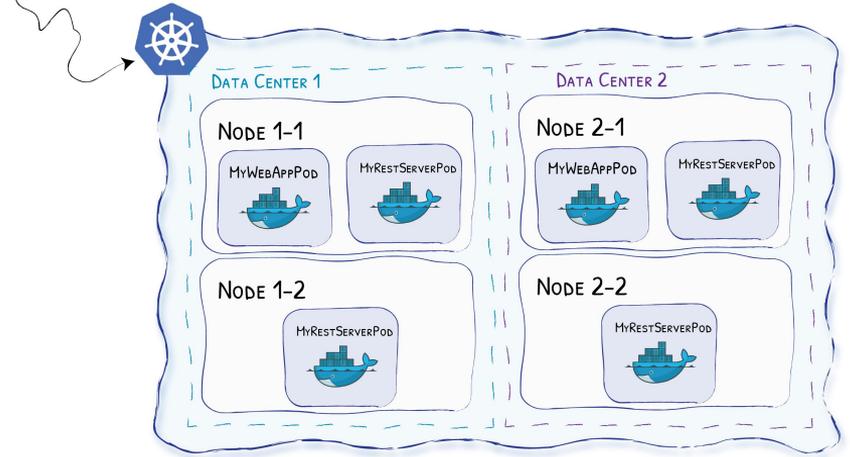
A single Pod can hold multiple containers.



K8s 101 - Deployments

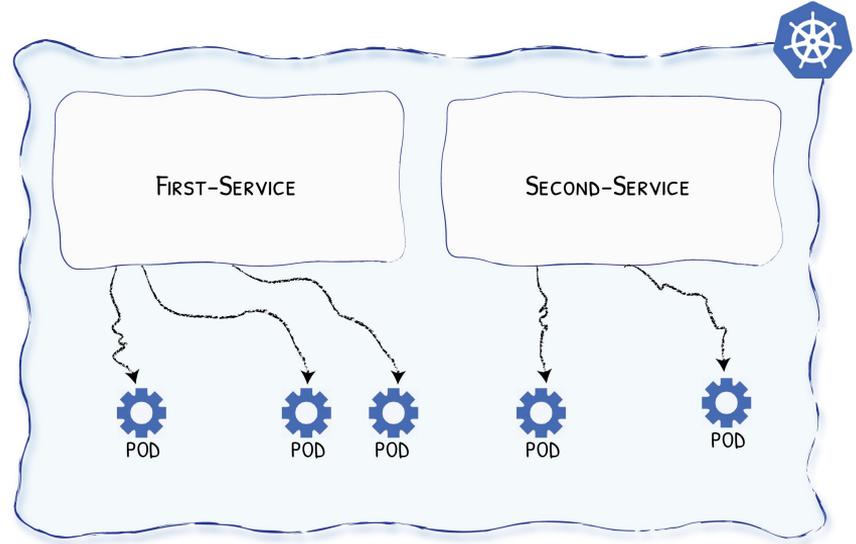
Deployments are requirements you give to Kubernetes regarding your applications (Pods)

I WANT 2 REPLICAS FOR MyWEBAPP
AND 4 REPLICAS FOR MyRETSERVER

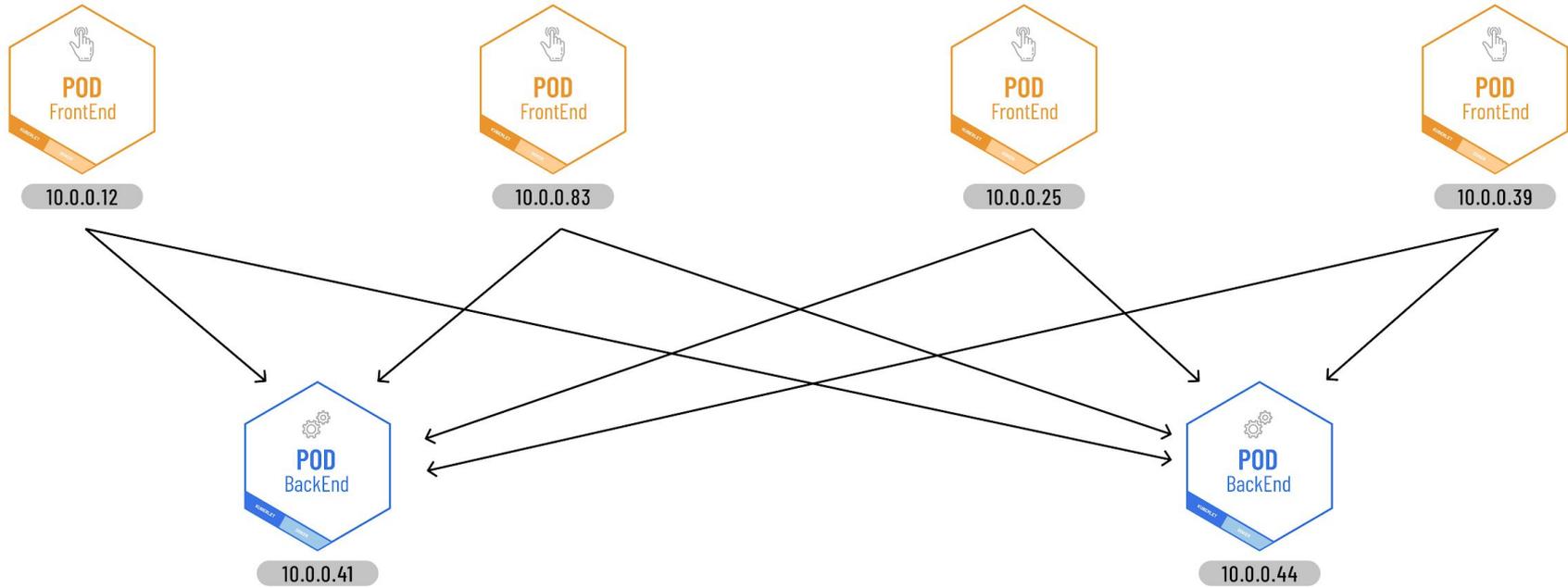


K8s 101 - Services

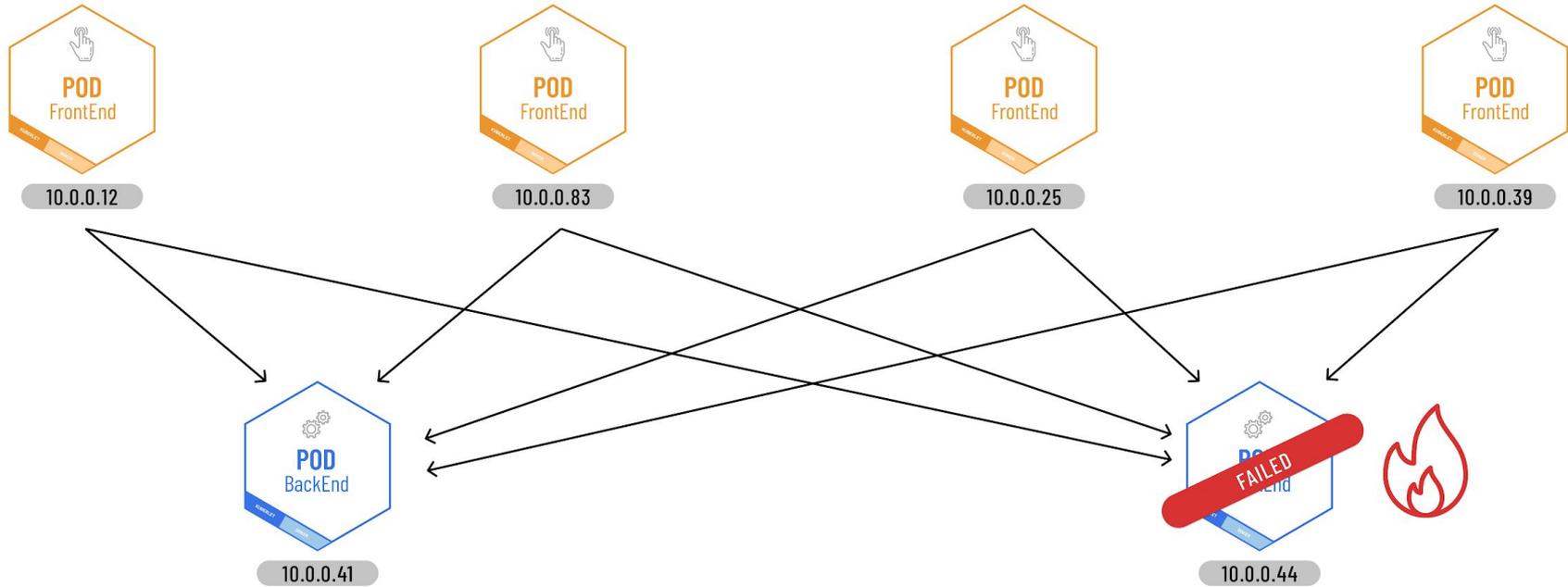
Services are an abstract way to expose an application running on a set of Pods as a network service.



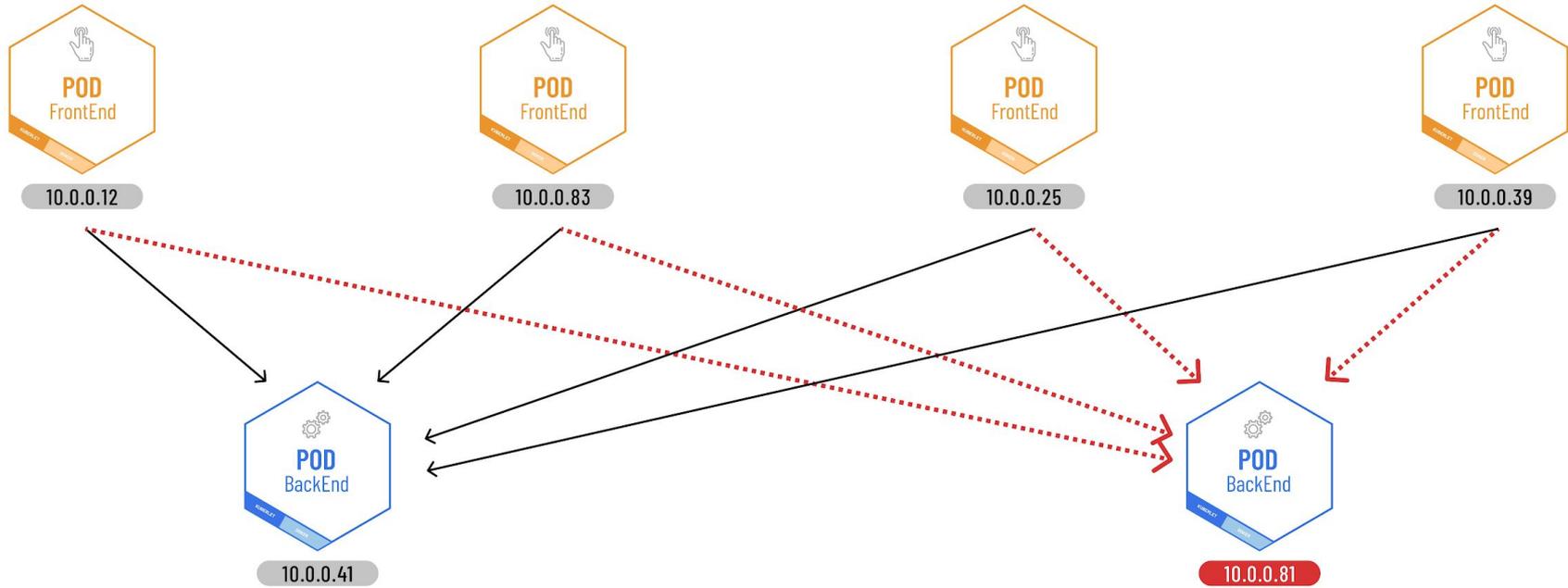
K8s 101 - Services



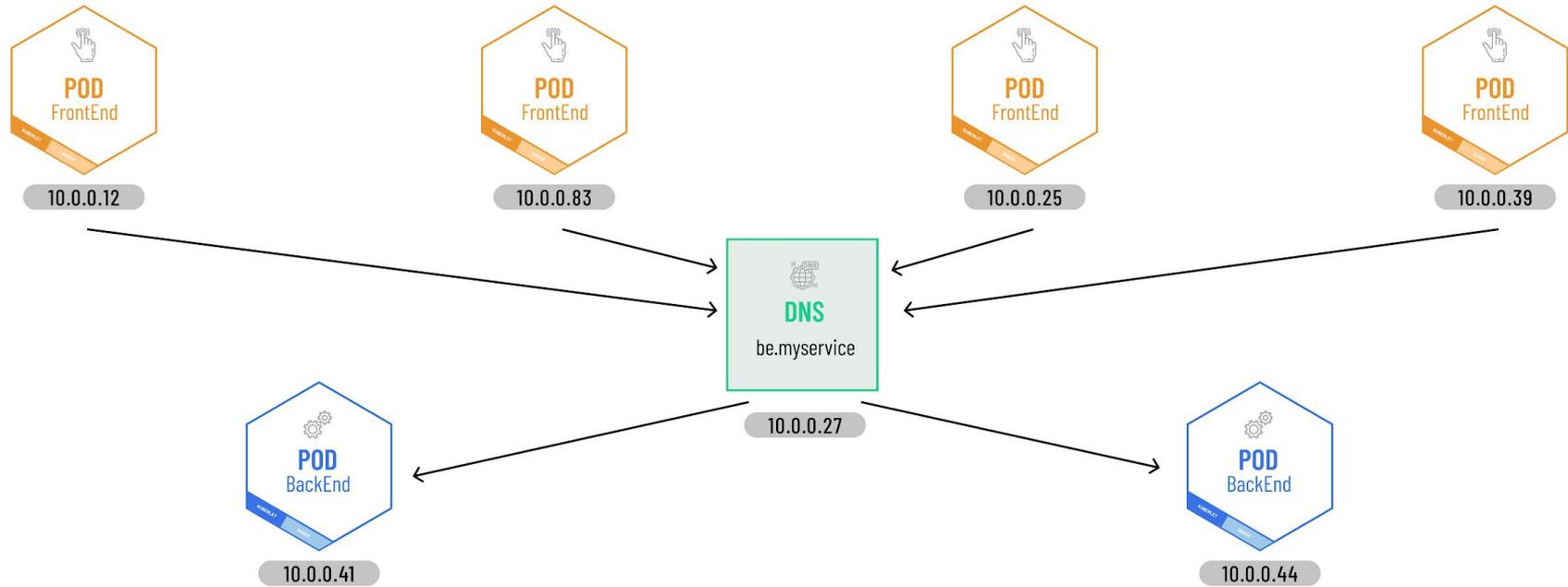
K8s 101 - Services



K8s 101 - Services



K8s 101 - Services





Declarative model & Desired state



Management techniques

The **kubectl** command-line tool supports several different ways to create and manage Kubernetes objects:

- Imperative commands
- Imperative object configuration
- Declarative object configuration



Imperative commands

The simplest way to get started or to run a one-off task in a cluster.

```
kubectl run nginx --image nginx
```



Imperative object configuration

In imperative object configuration, the `kubectl` command specifies the operation (create, replace, etc.), optional flags and at least one file name.

The file specified must contain a full definition of the object in YAML or JSON format.

```
kubectl create -f nginx.yaml
```



Declarative object configuration

Using declarative object configuration, a user operates on object configuration files stored locally, however the user does not define the operations to be taken on the files.

Create, update, and delete operations are automatically detected per-object by kubectl.

```
kubectl apply -f configs/
```

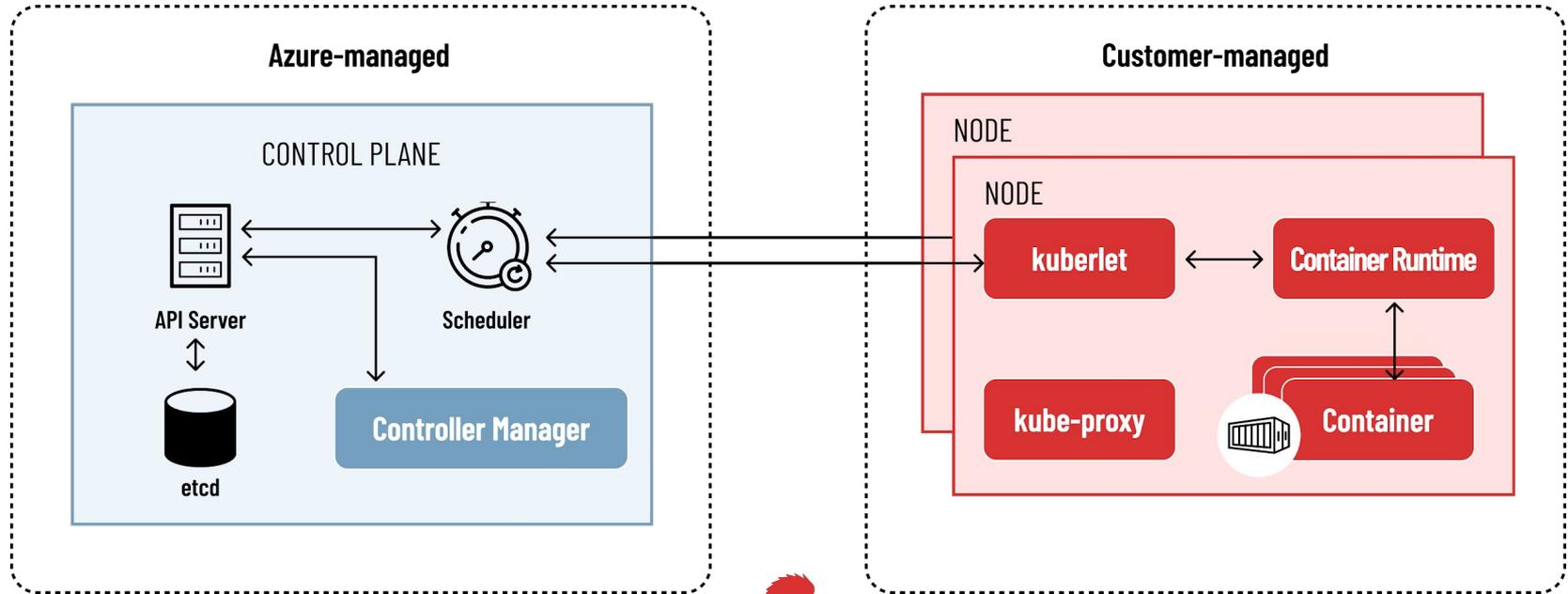


K8s + Azure = AKS



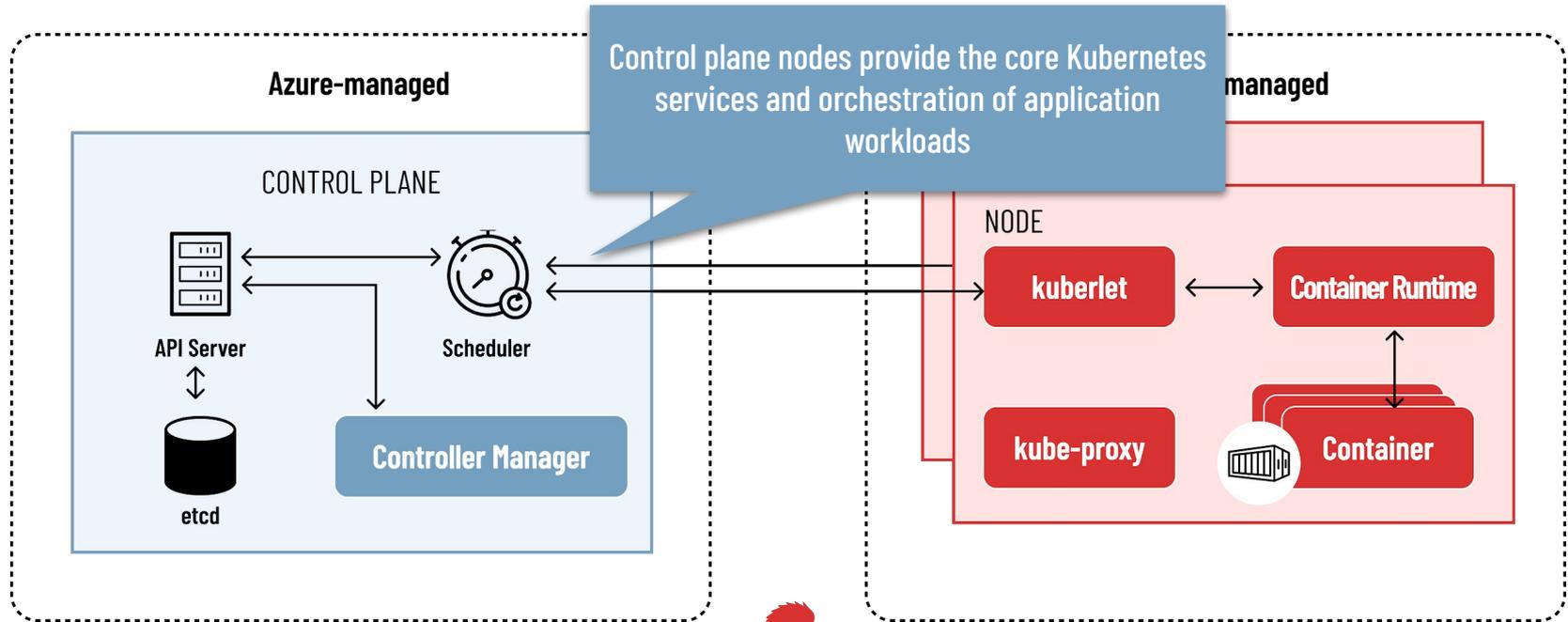
Kubernetes cluster architecture

A Kubernetes cluster is divided into two components:



Kubernetes cluster architecture

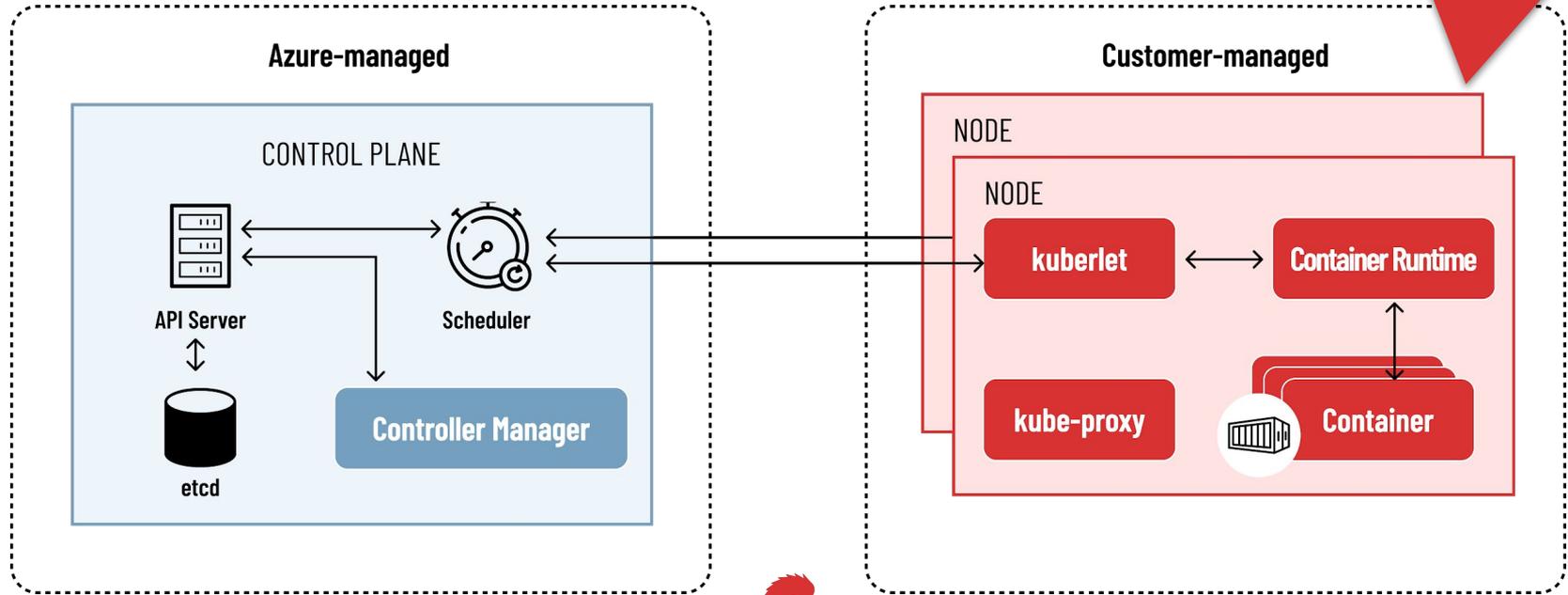
A Kubernetes cluster is divided into two components:

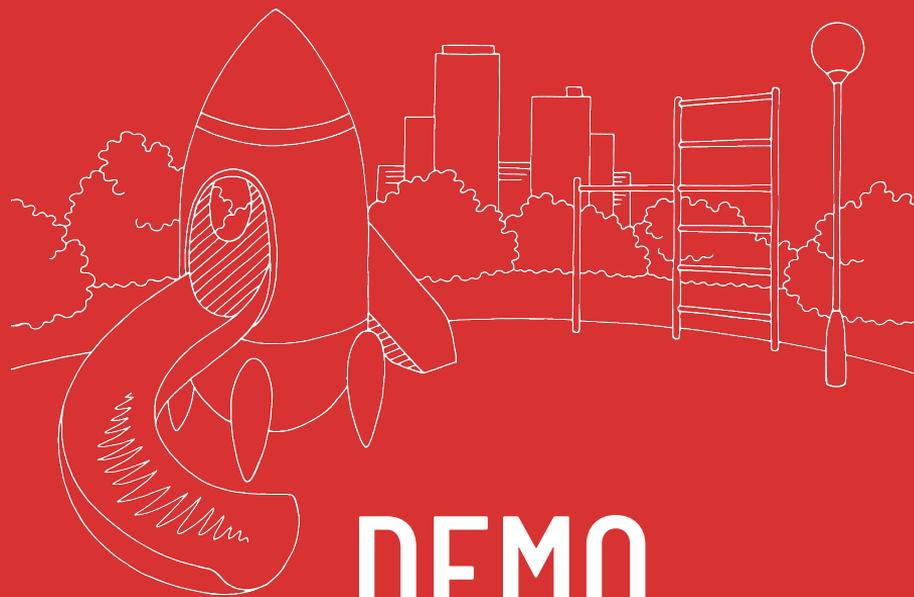


Kubernetes cluster architecture

A Kubernetes cluster is divided into two components:

Nodes run your application workloads.





DEMO





Microservice architecture

THE "GOOD"

- An application is sum of its components
- Better fault isolation
- Components can be spread across multiple servers

THE "BAD"

- Many components, many moving parts
- Difficult to manage inter-communication
- **Manual management can be difficult**





ALESSANDRO MELCHIORI

Founder & Software developer @CodicePlastico
alessandro@codiceplastico.com
@amelchiori



alessandro@codiceplastico.com
@amelchiori

Slide e materiale su
<https://aspit.co/netconfit-20>

.NET Conference
Italia 2020

.NET