

Provide and Troubleshoot Access to Applications via Services



Nigel Poulton

Author & Trainer

@nigelpoulton nigelpoulton.com



Agenda



Understanding Kubernetes Services

Working with Kubernetes Services

Exam Scenarios

Recap and Test Yourself



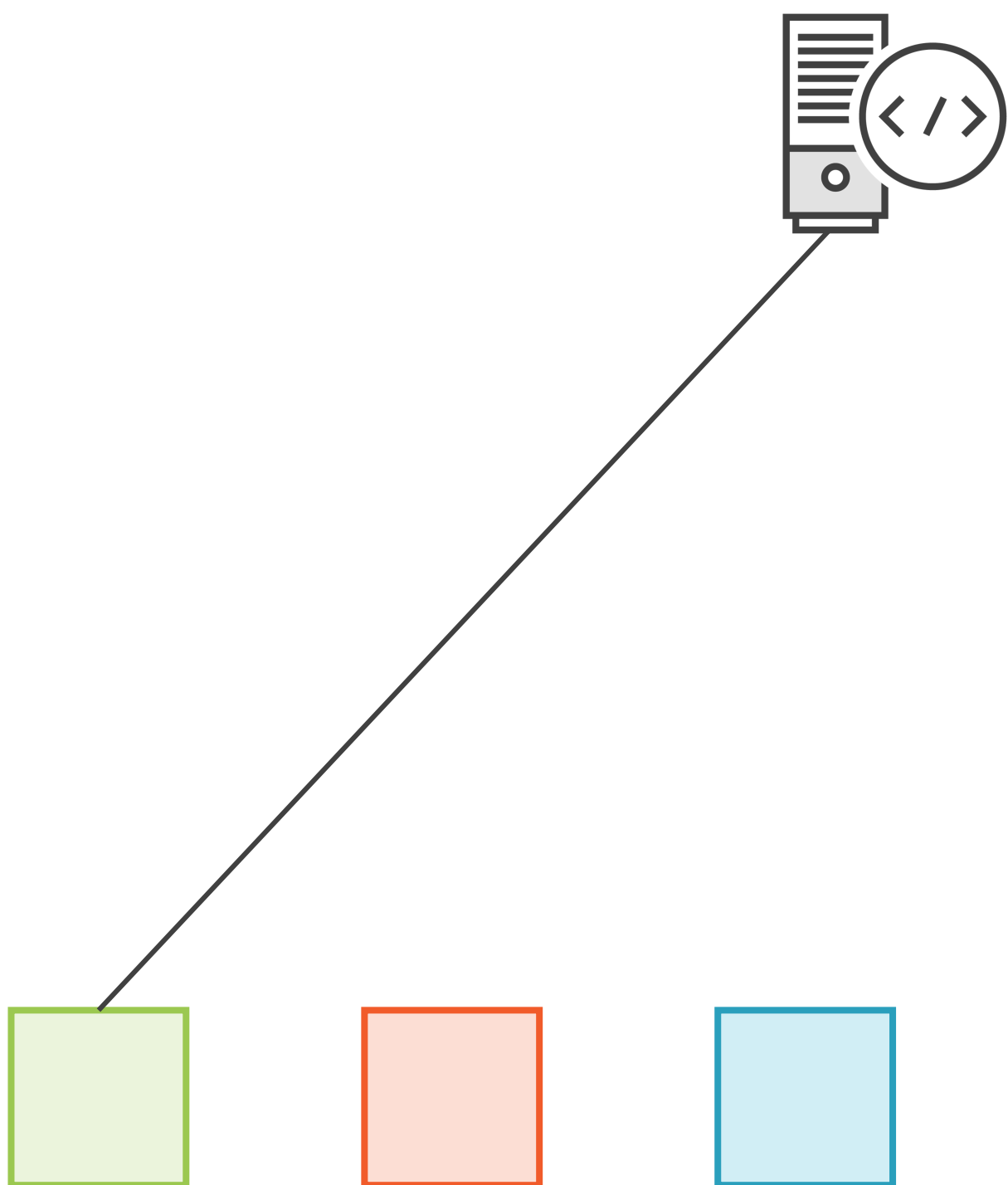
Understanding Kubernetes Services

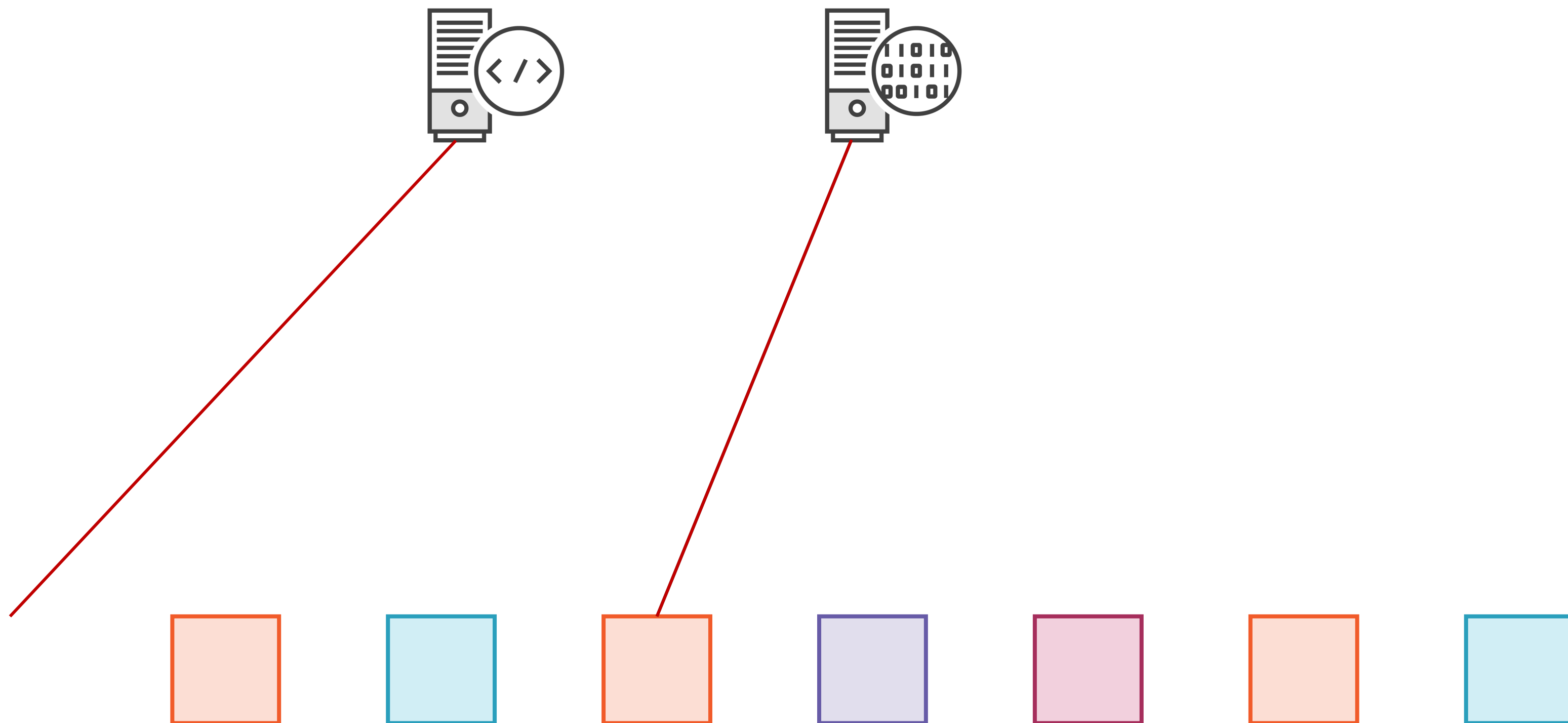


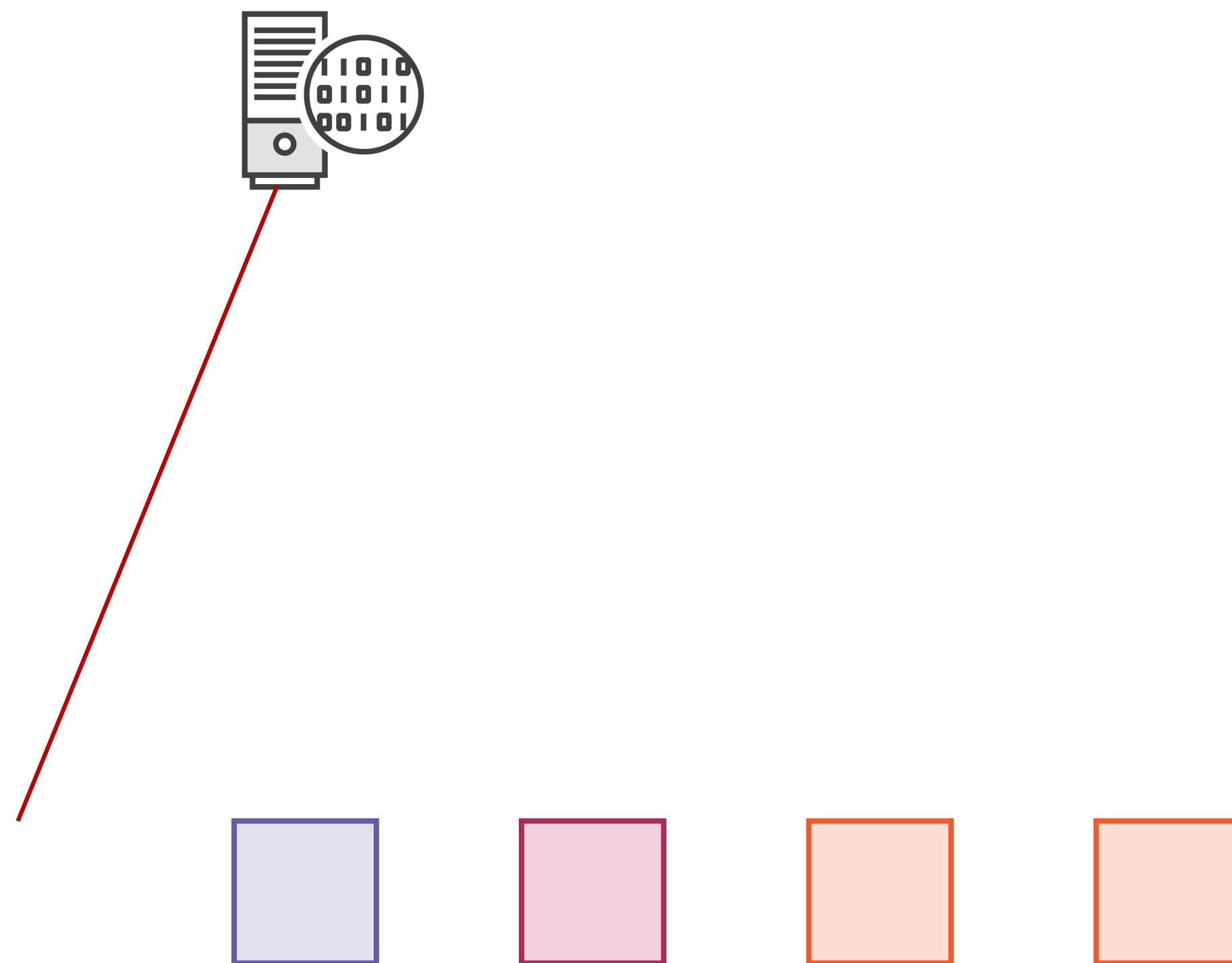
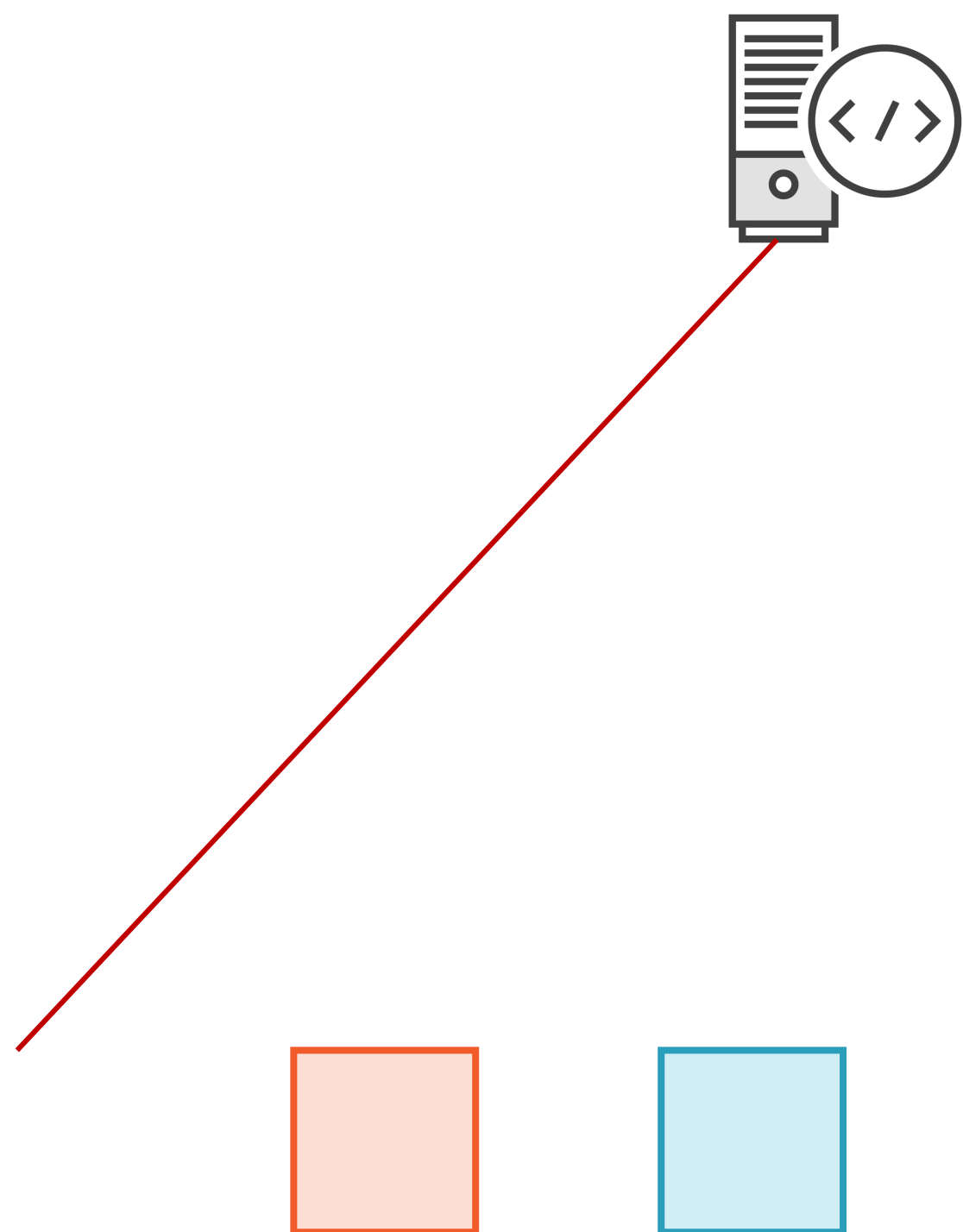
Service

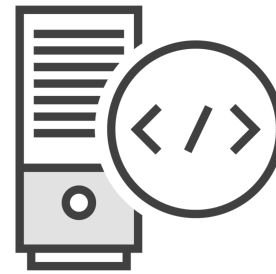
Kubernetes network abstraction as defined in the v1 core API group.



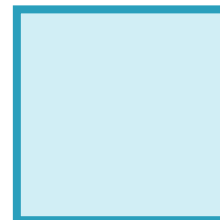
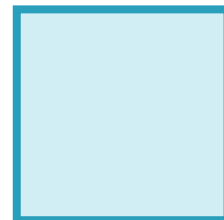


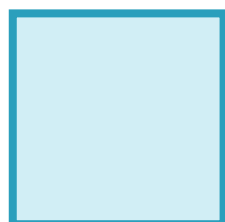
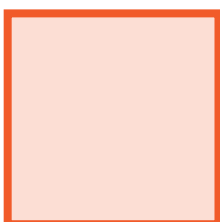
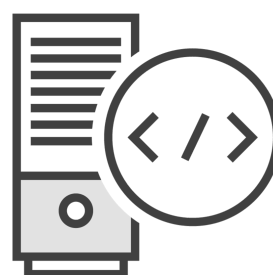


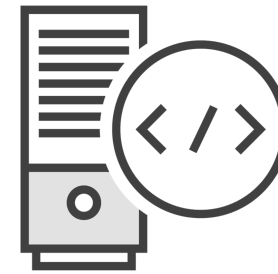




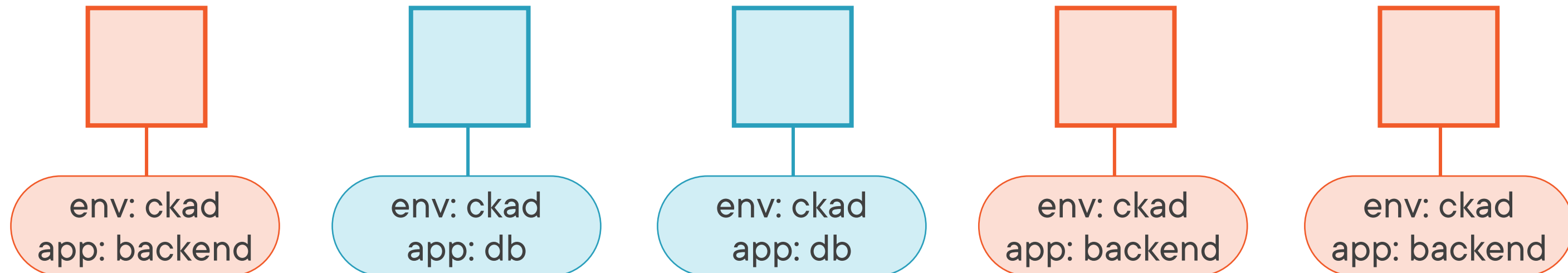
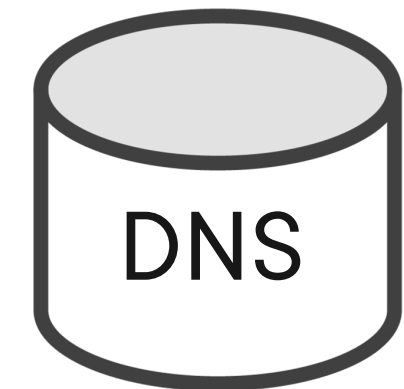
Service

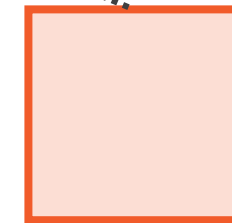
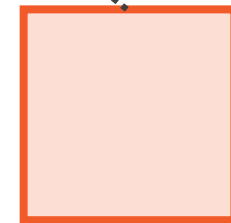
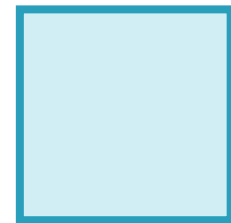
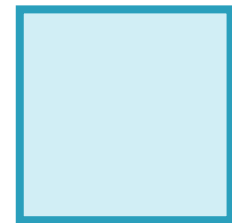
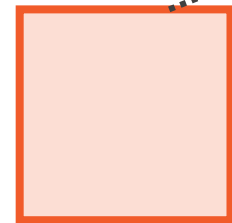
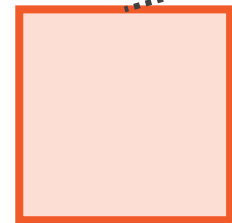
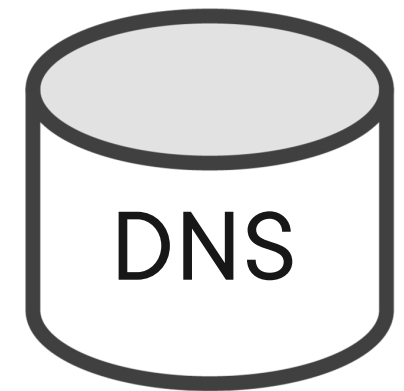
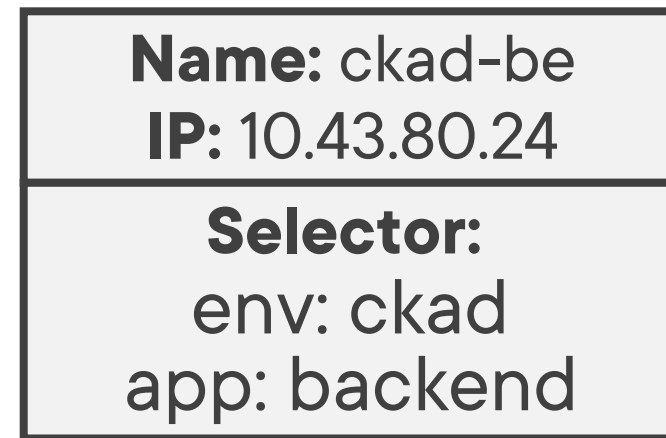
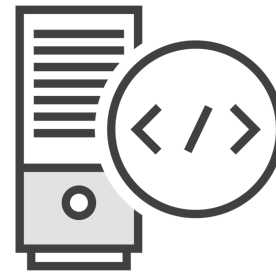






Name: ckad-be IP: 10.43.80.24
Selector: env: ckad app: backend





env: ckad
app: backend

env: ckad
app: backend

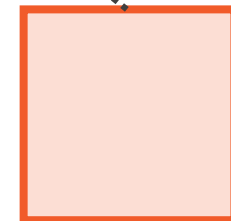
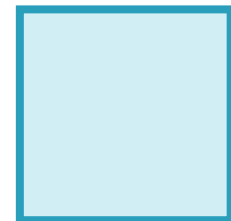
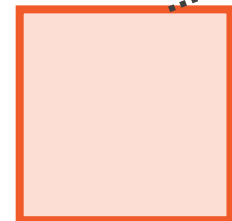
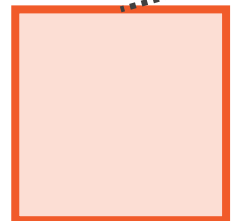
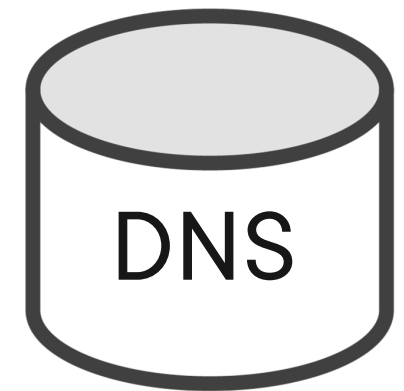
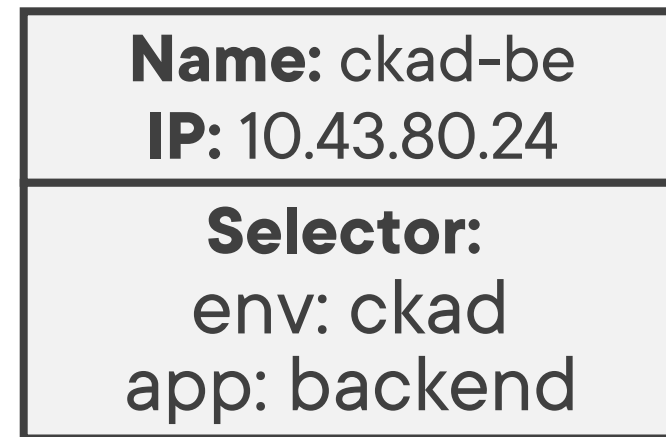
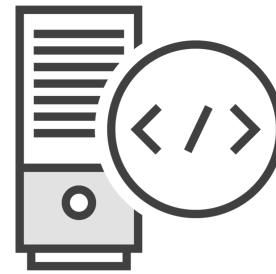
env: ckad
app: db

env: ckad
app: db

env: ckad
app: backend

env: ckad
app: backend





env: ckad
app: backend

env: ckad
app: backend

env: ckad
app: db

env: ckad
app: db

env: ckad
app: backend

env: ckad
app: backend



ClusterIP

**Internal IP and port on
the Pod network**

NodePort

**Static port on all
cluster nodes**

LoadBalancer

**Integrate with cloud
load-balancer**



Exam Tips

Services

Connect to Services
instead of Pods

ClusterIP

For internal traffic

NodePort

Expose on static port
on all nodes

LoadBalancer

Expose via cloud
load-balancer

Matching

Label selector must
match Pod labels



Working with Kubernetes Services



Exam Scenarios



← Prev.

☑️☐☐ Task 1 of X

Next →

Task weight: 6%



Cluster: ckad0014

Namespace: dev

Doc links: Services, Pods, Deployments

Your developers have designed an app with two microservices called **store-backend** and **store-db**. They're defined in **backend.yml** and **db.yml** and both can scale to be multiple Pods.

The backend Pods need a reliable way to connect to the **store-db** microservice via a DNS name called **db** on **TCP 5432**.

Both microservices are deployed to the dev Namespace. However, the backend microservice has not started as the **db** DNS endpoint has not been created.

Task

Create and deploy the highly available **db** network endpoint. When successfully deployed, the backend microservice will start.



← Prev.

☒ ☐ ☐ Task 2 of X

Next →

Task weight: 8%



Cluster: ps-ckad
Namespace: pluralsight
Doc links: Services, Pods, Deployments

There's an app called **web-fe** defined in the **app.yml** file in your working directory.

It's designed to deploy 5 replicas that are accessible on **port 8080** via the **Service called web**. However, the Service is not sending traffic to the replicas.

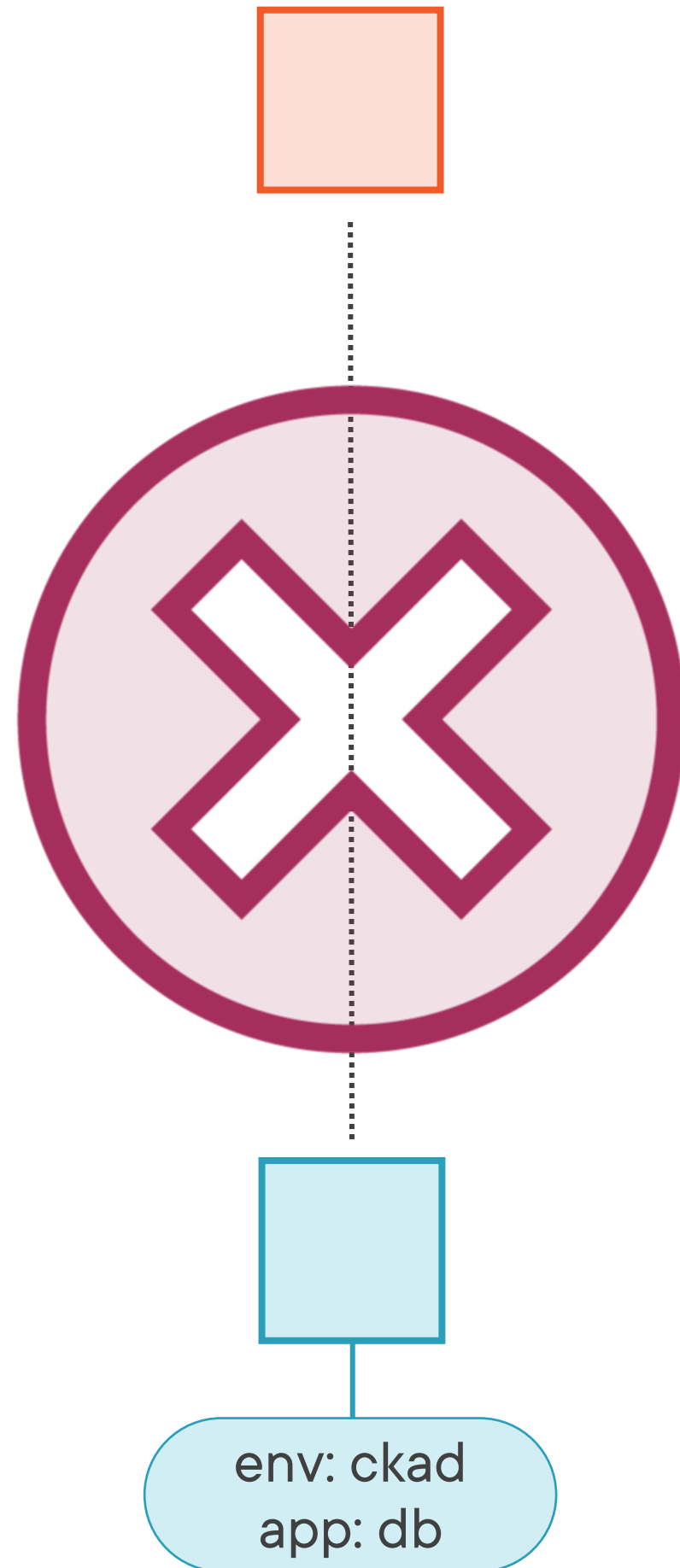
Task

Troubleshoot and resolve the problem.



Recap and Test Yourself





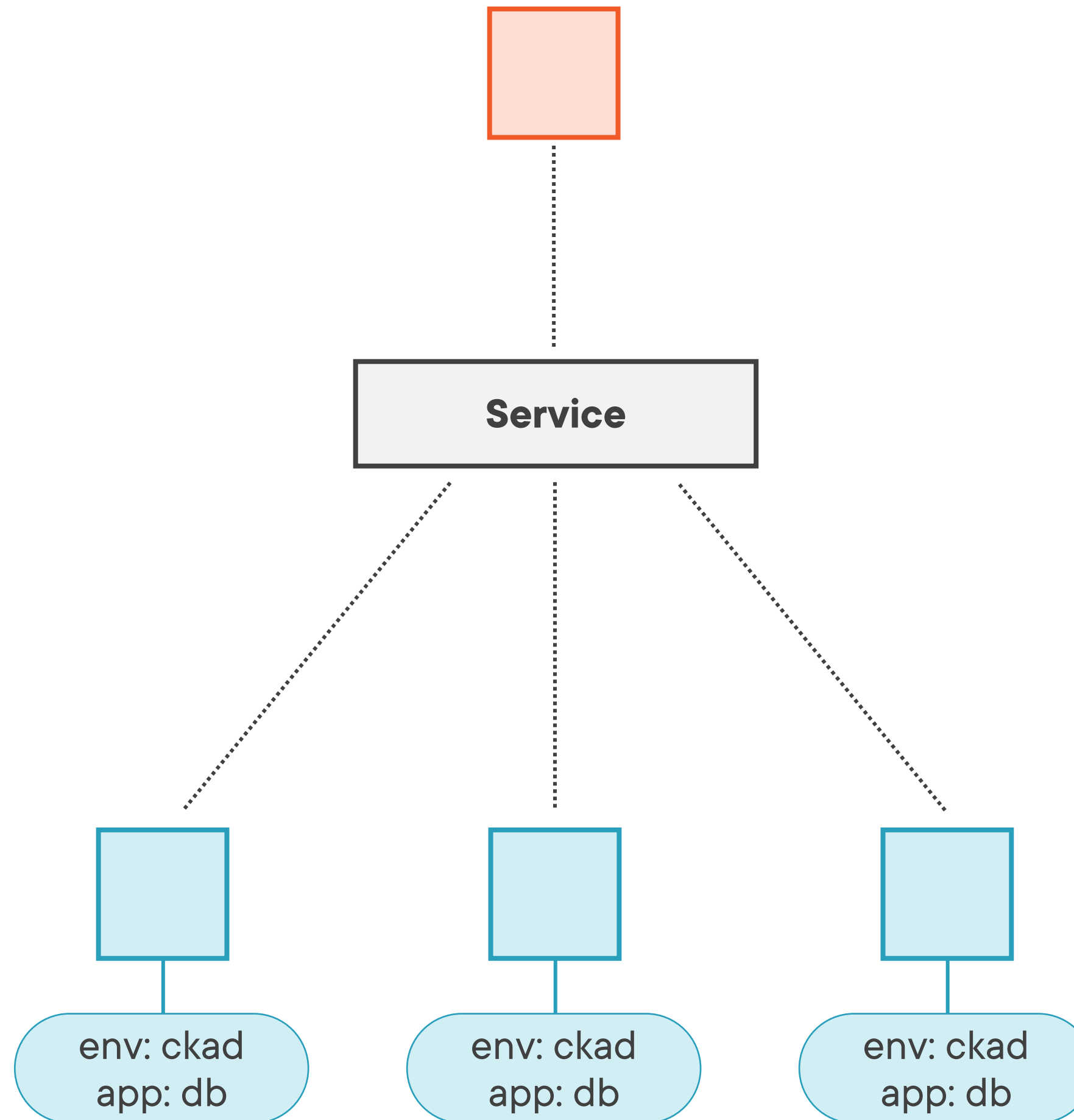
LoadBalancer

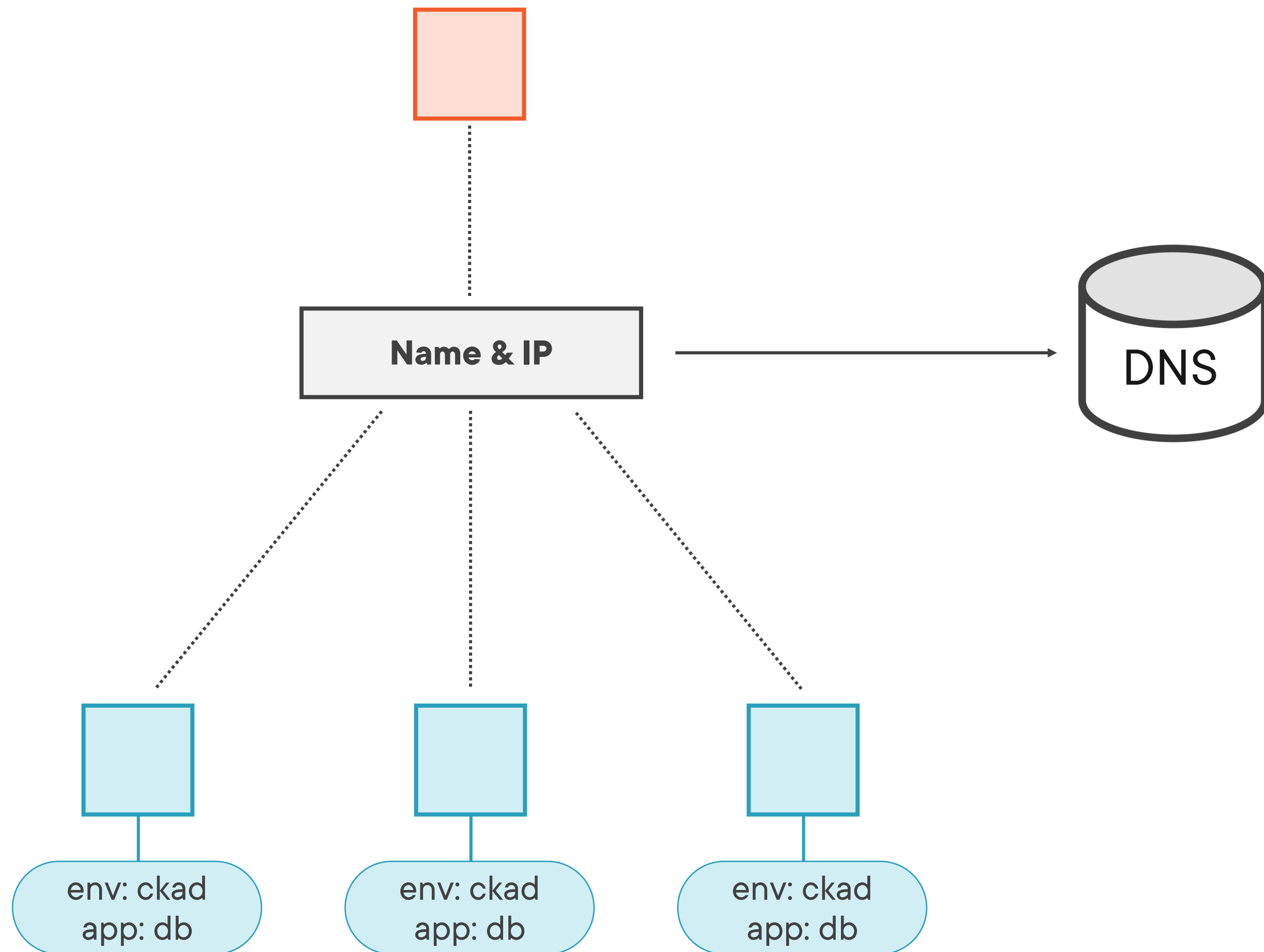
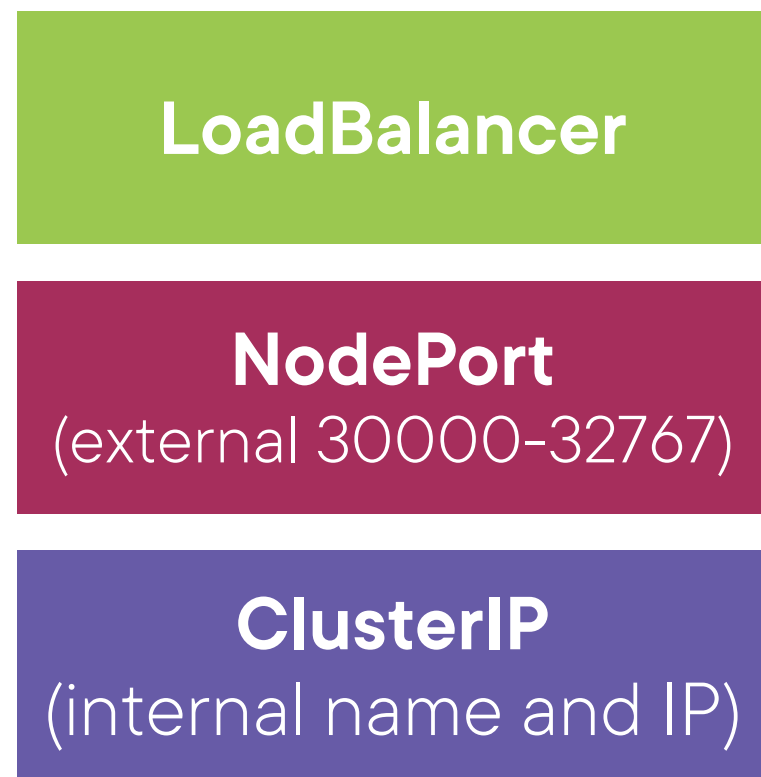
NodePort

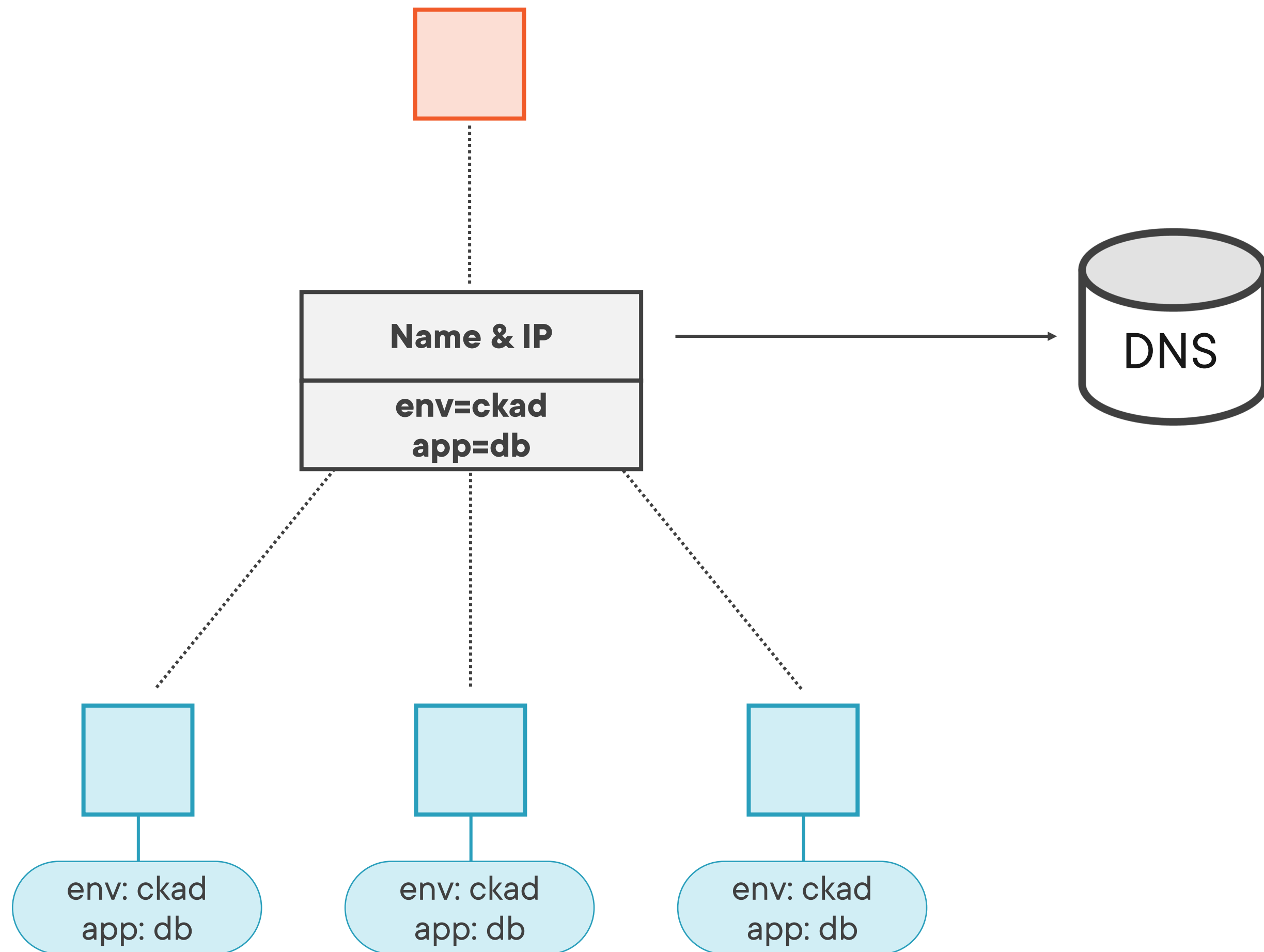
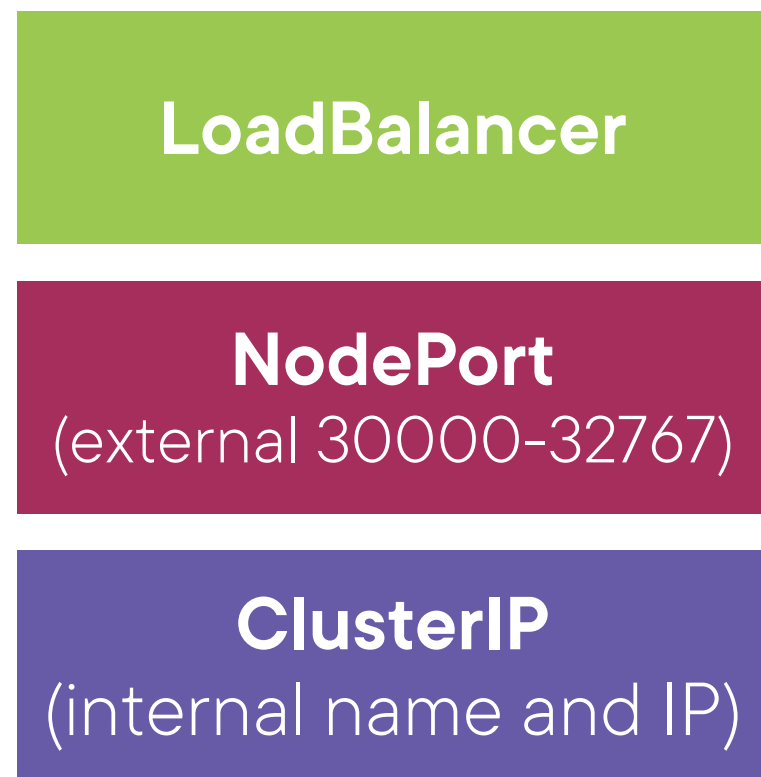
(external 30000-32767)

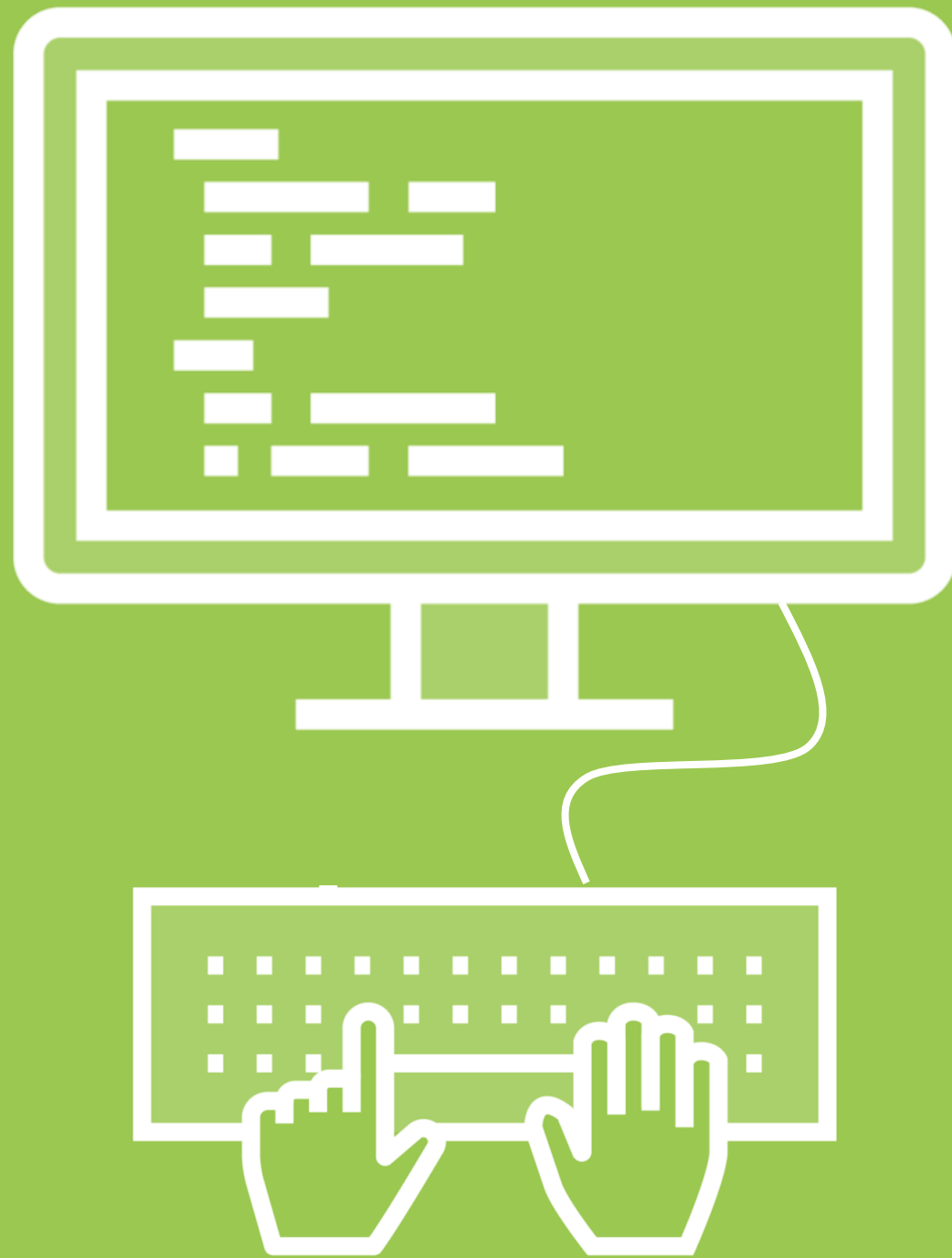
ClusterIP

(internal name and IP)









GitHub Repo

<https://github.com/nigelpoulton/ckad>

Navigate to:

- 5 Services and Networking
- 3 Provide and Troubleshoot Access to Apps via Services



Up Next:

Use Ingress Rules to Expose Applications

