KUBERNETES 101

DEPLOYING YOUR FIRST CLOUD NATIVE APPLICATION

CATALIN JORA | CONTAINER SOLUTIONS | @JOCATALIN

ATMOSPHERE CONF KRAKOW | 16 MAY 2017

WHO AM I

Catalin Jora (Engineer @ Container Solutions)
@jocatalin

KUBERNETES 101

Why: cloud native applications
What: the building blocks
How: deploying application (demo)

WHY KUBERNETES

defining cloud native applications

SOFTWARE CAPABILITIES WISH LIST 2017

Speed

Scale

Costs

SPEED OF RELEASE

Release cycle of software will only acelerate, will (probably) never slow down.

SCALE

Software should be able to grow with user/business demand

COSTS

Reduce infrastructure costs by moving from pay in advance to pay per use

CLOUD NATIVE SYSTEMS

distributed systems environments capable of scaling to tens of thousands of self healing multi-tenant nodes | cncf

CLOUD NATIVE APPLICATIONS

Packed in containers

Dinamically managed

Micro-service oriented

WHAT IS KUBERNETES

Kubernetes is a production-grade, open-source platform that orchestrates the placement (scheduling) and execution of application containers within and across computer clusters.

KUBERNETES CAN RUN ON

laptop

public cloud

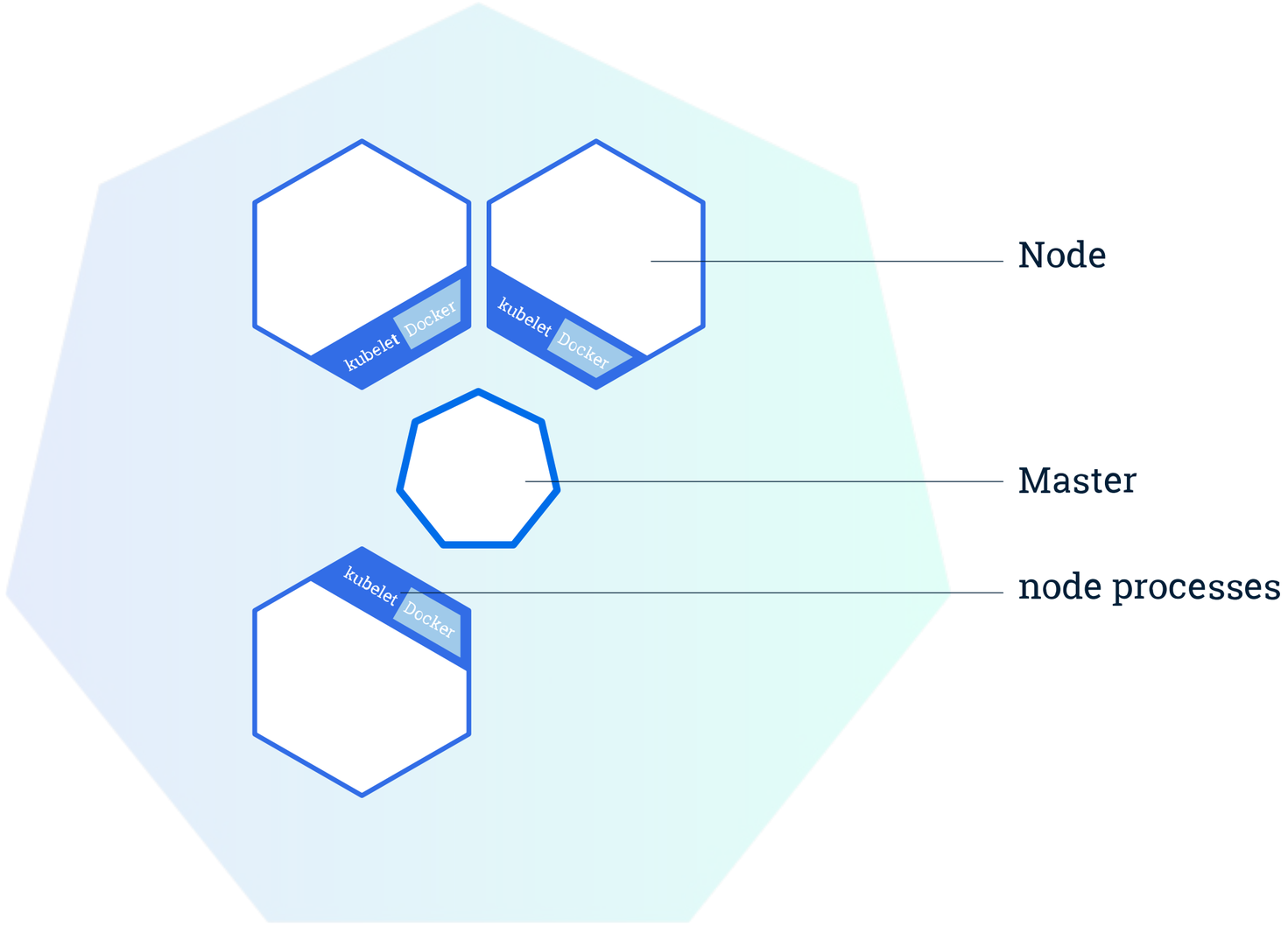
on-premises

HOW TO RUN KUBERNETES

vanilla (you will do the maintenance)

managed (as a service)

KUBERNETES ARCHITECTURE



KUBERNETES ARCHITECTURE

MASTER

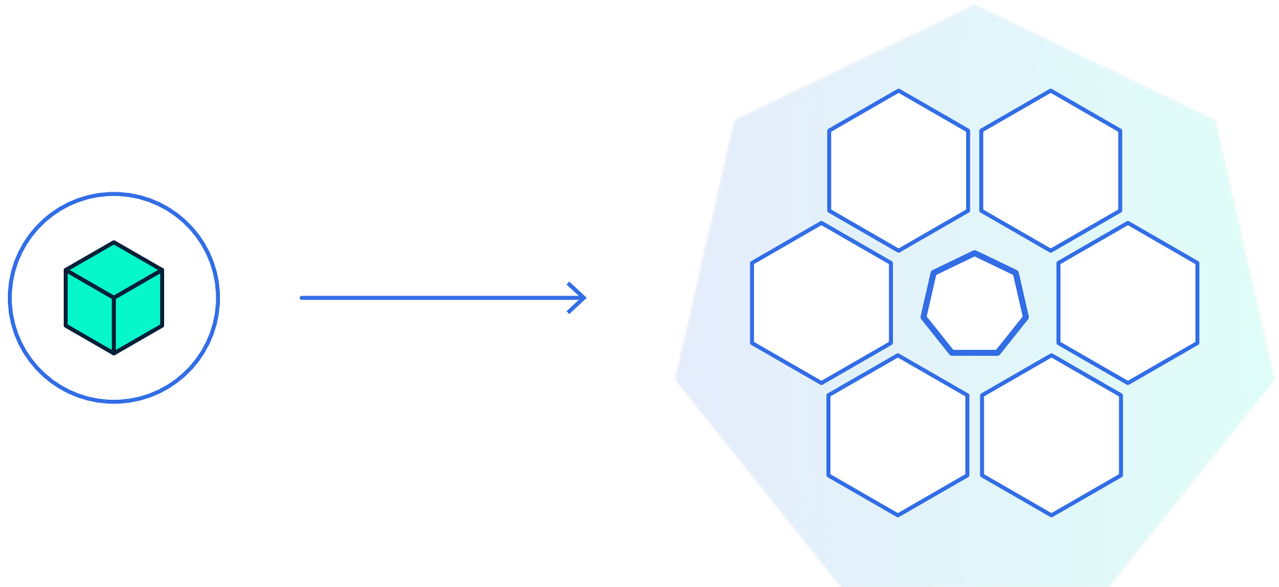
etcd, API server, controller manager, scheduler, kubelet

NODES

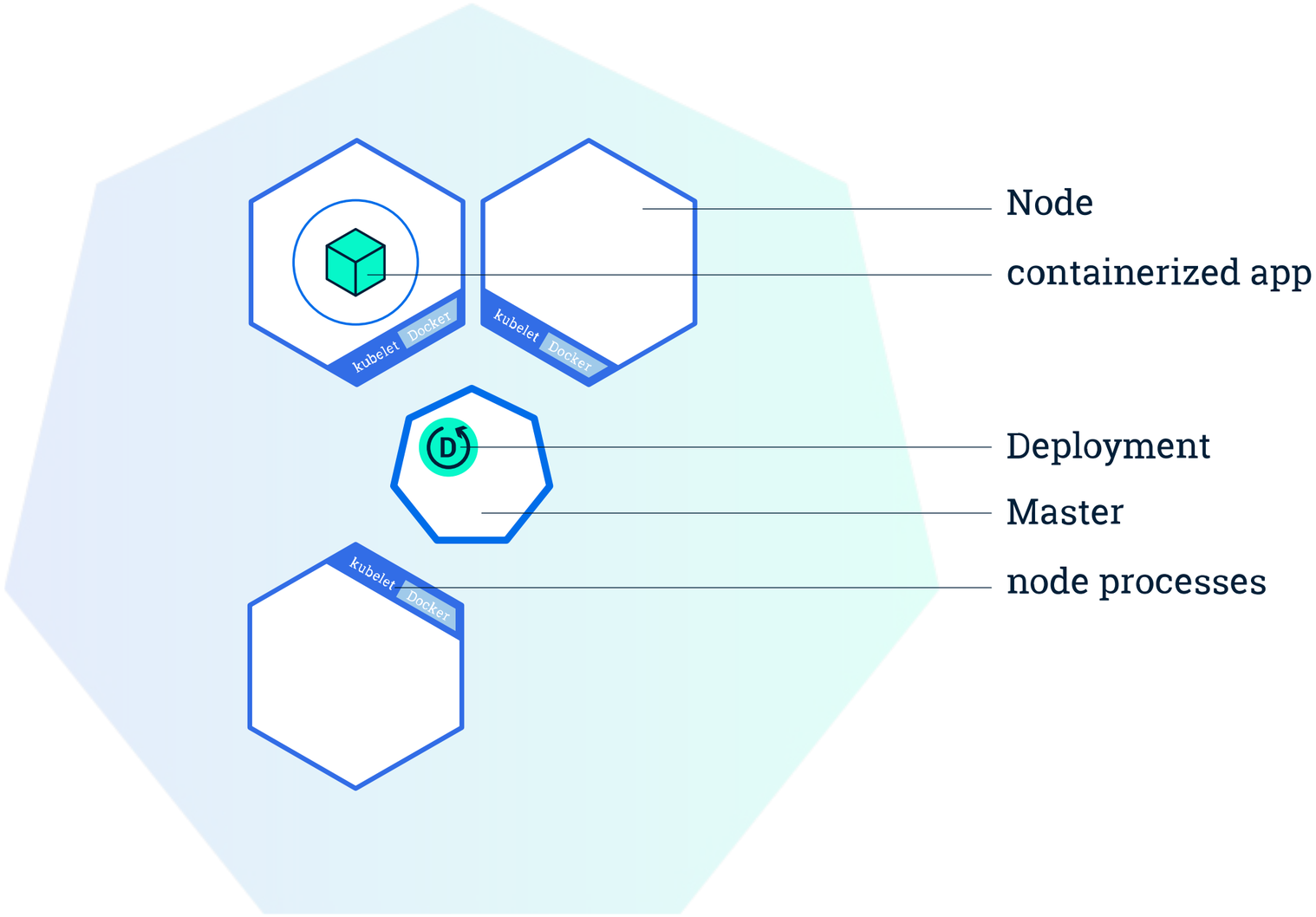
kubelet, docker

KUBERNETES DEPLOYMENT

- responsible for creating and updating the instances of your applications
- provide a self-healing mechanism

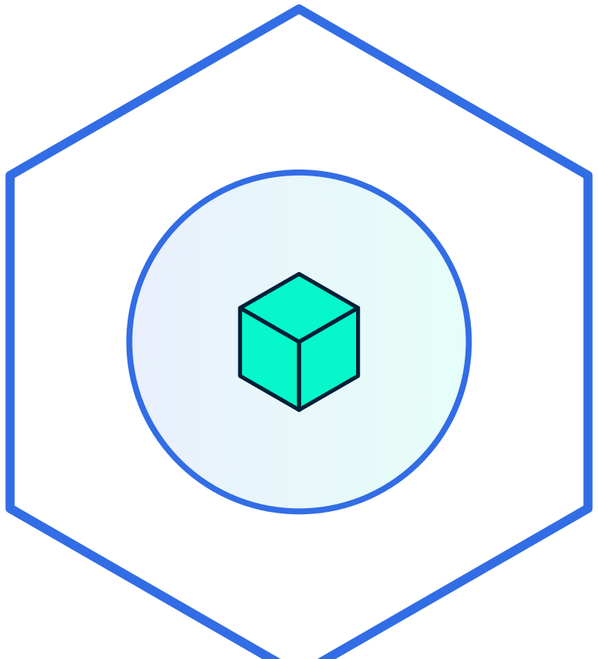


KUBERNETES DEPLOYMENT



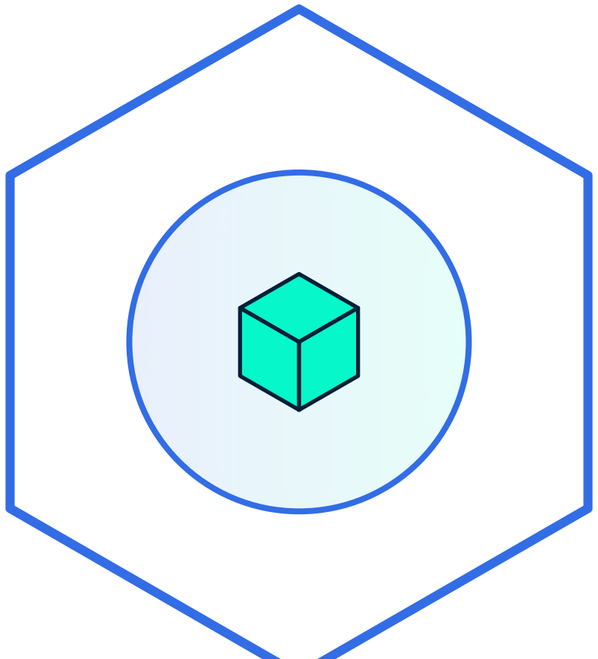
KUBERNETES POD

- a group of one or more containers (e.g. Docker)
- shared storage & unique cluster level IP
- info about container image, ports, resources (CPU,RAM)

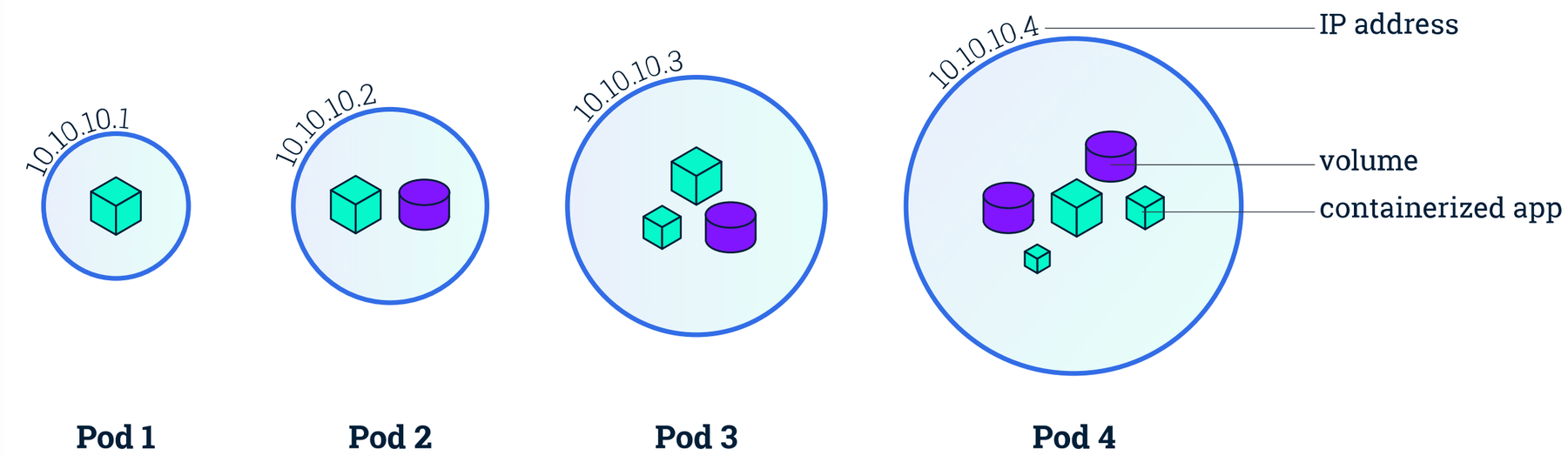


KUBERNETES POD

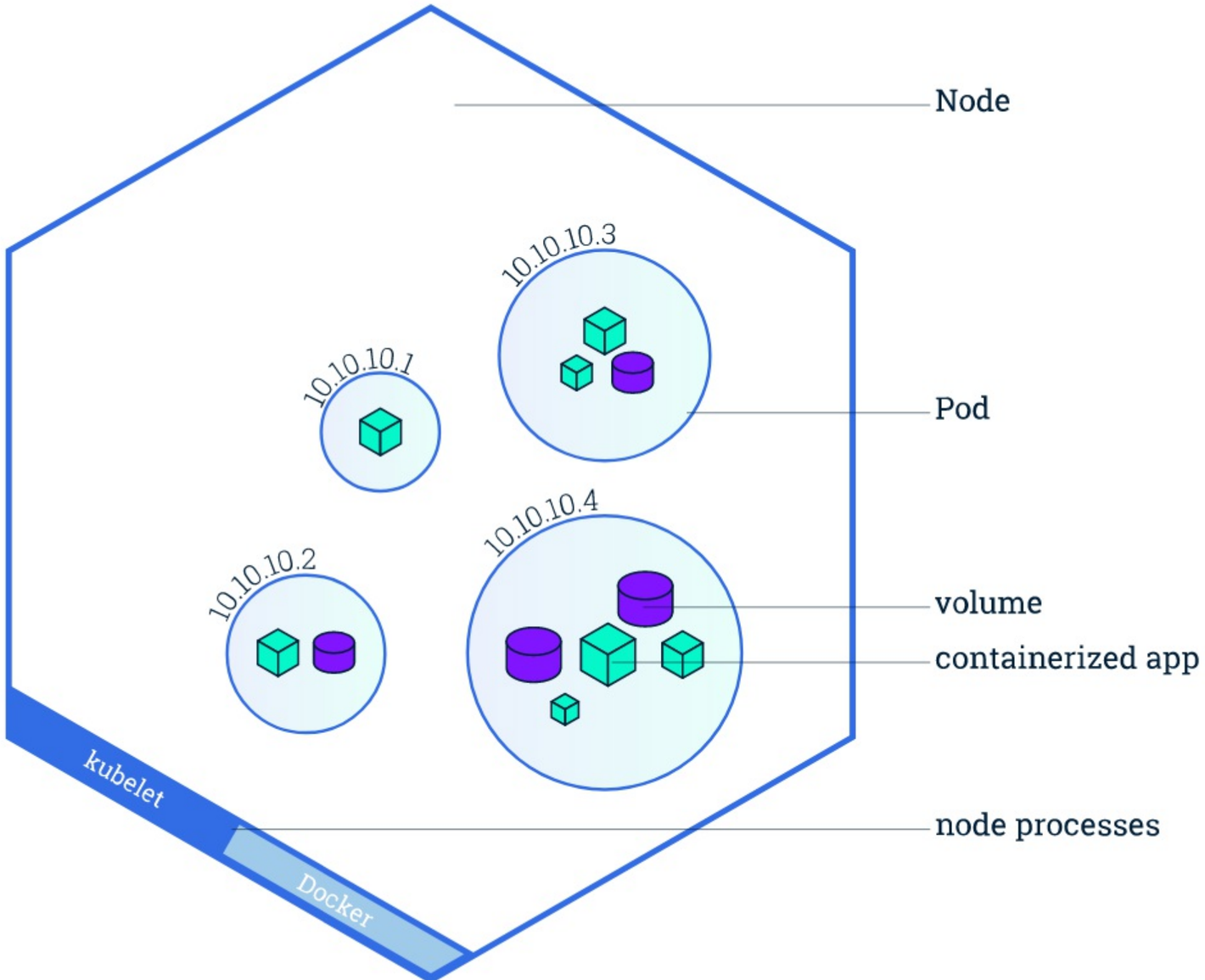
- are mortal
- are the atomic unit on Kubernetes
- deployments are creating pods with containers inside them



KUBERNETES POD



KUBERNETES NODE

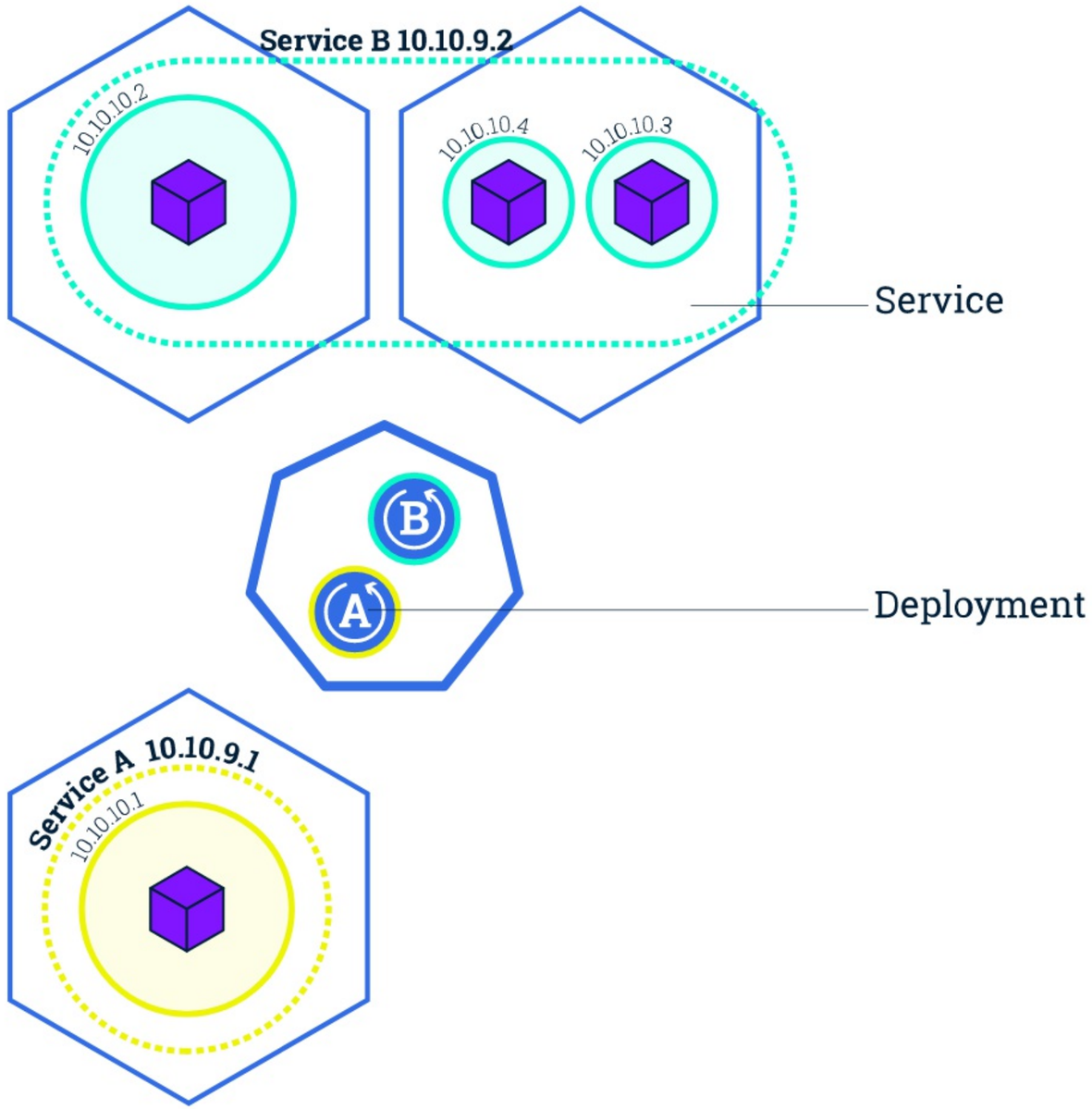


KUBERNETES SERVICE

-

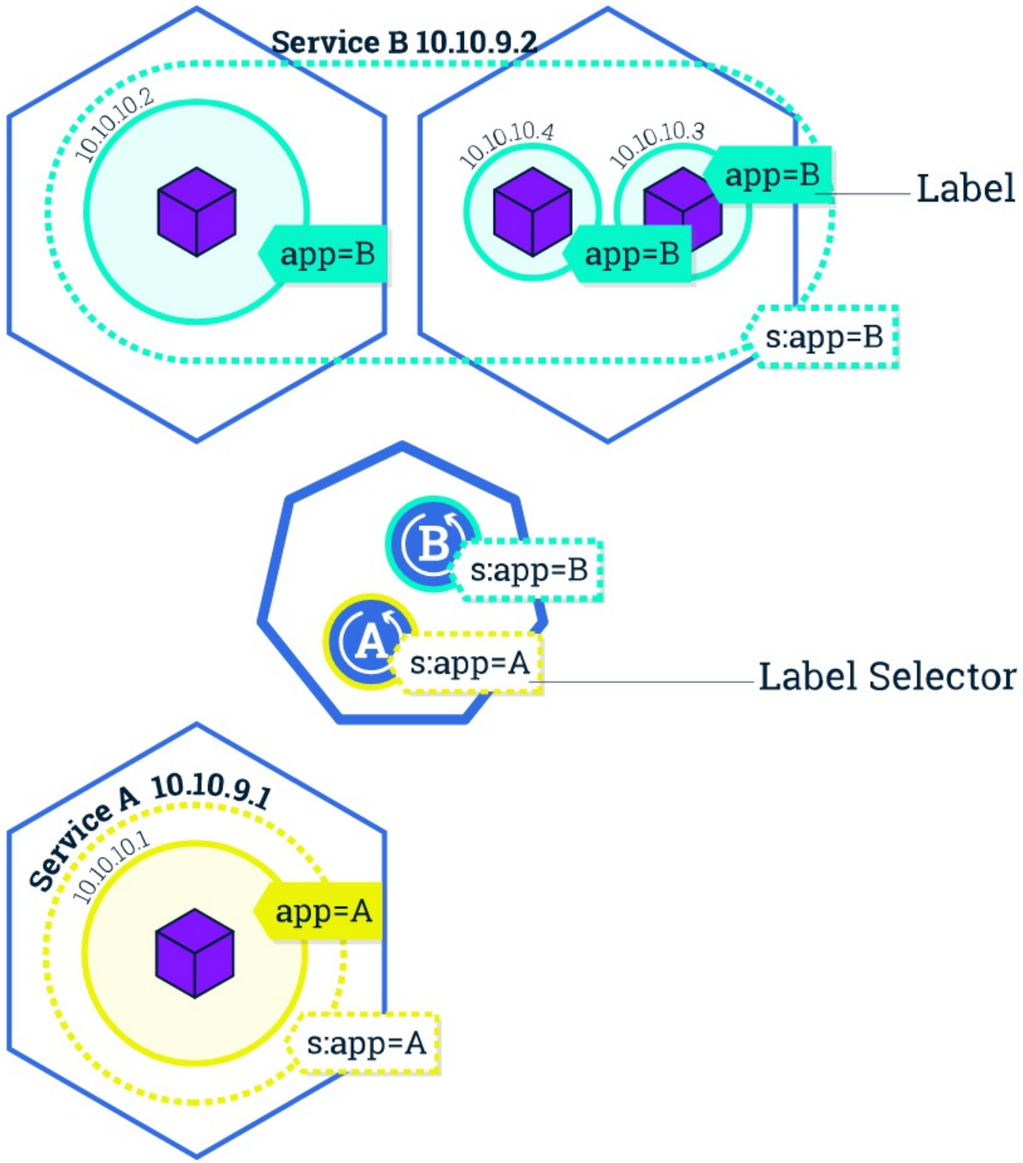
ABSTRACTION LAYER WHICH DEFINES A LOGICAL SET OF PODS

- enables external traffic exposure
- load balancing
- service discovery for pods


Service B 10.10.9.2
10.10.10.2
10.10.10.4
10.10.10.3
Service
B
A
Deployment
Service A 10.10.9.1
10.10.10.1

KUBERNETES LABEL

- key/value pairs attached to objects (E.g Pods)
- help organizing objects
- can be changed anytime

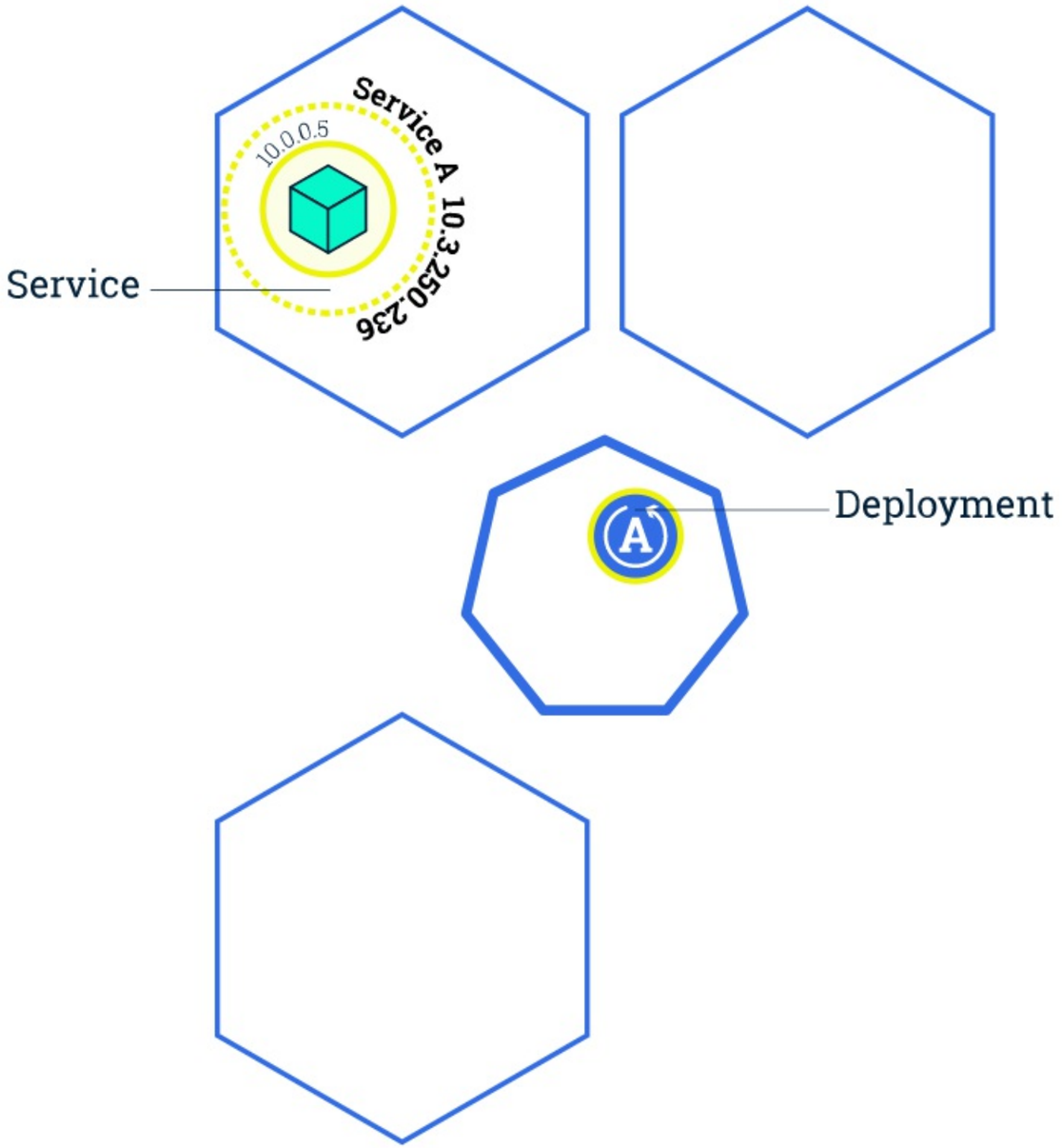

Service B 10.10.9.2
10.10.10.2
app=B
10.10.10.4
10.10.10.3
app=B
app=B
Label
s:app=B
B
s:app=B
A
s:app=A
Label Selector
Service A 10.10.9.1
10.10.10.1
app=A
s:app=A

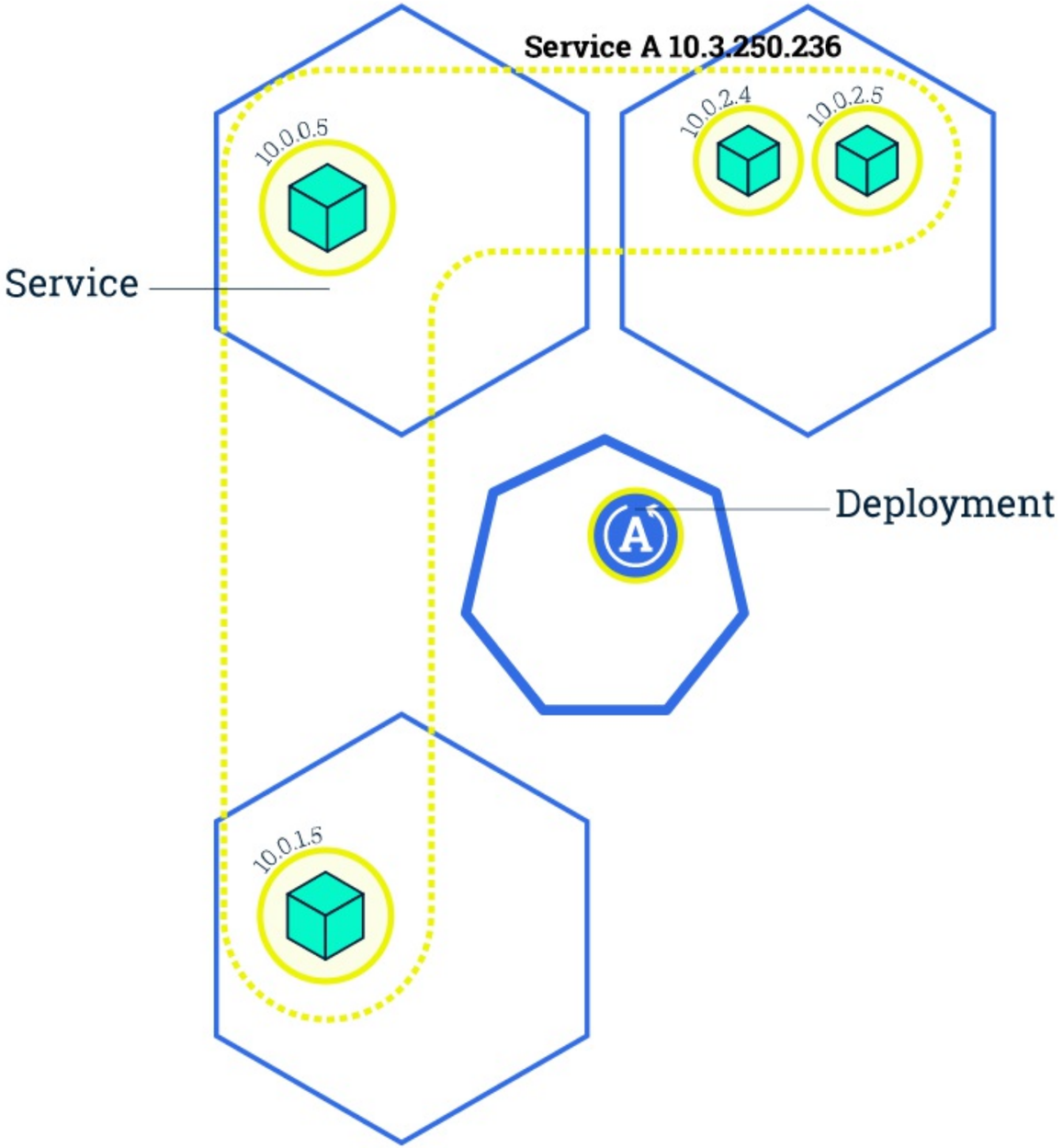
SCALLING APPLICATIONS

-

CHANGING THE NUMBER OF REPLICAS IN A DEPLOYMENT

- traffic is sent only to up & running pods
- new pods are created on nodes with available resources
- running multiple pod replicas allow zero-downtime updates


Service A 10.3.250.236
10.0.0.5
Service
Deployment

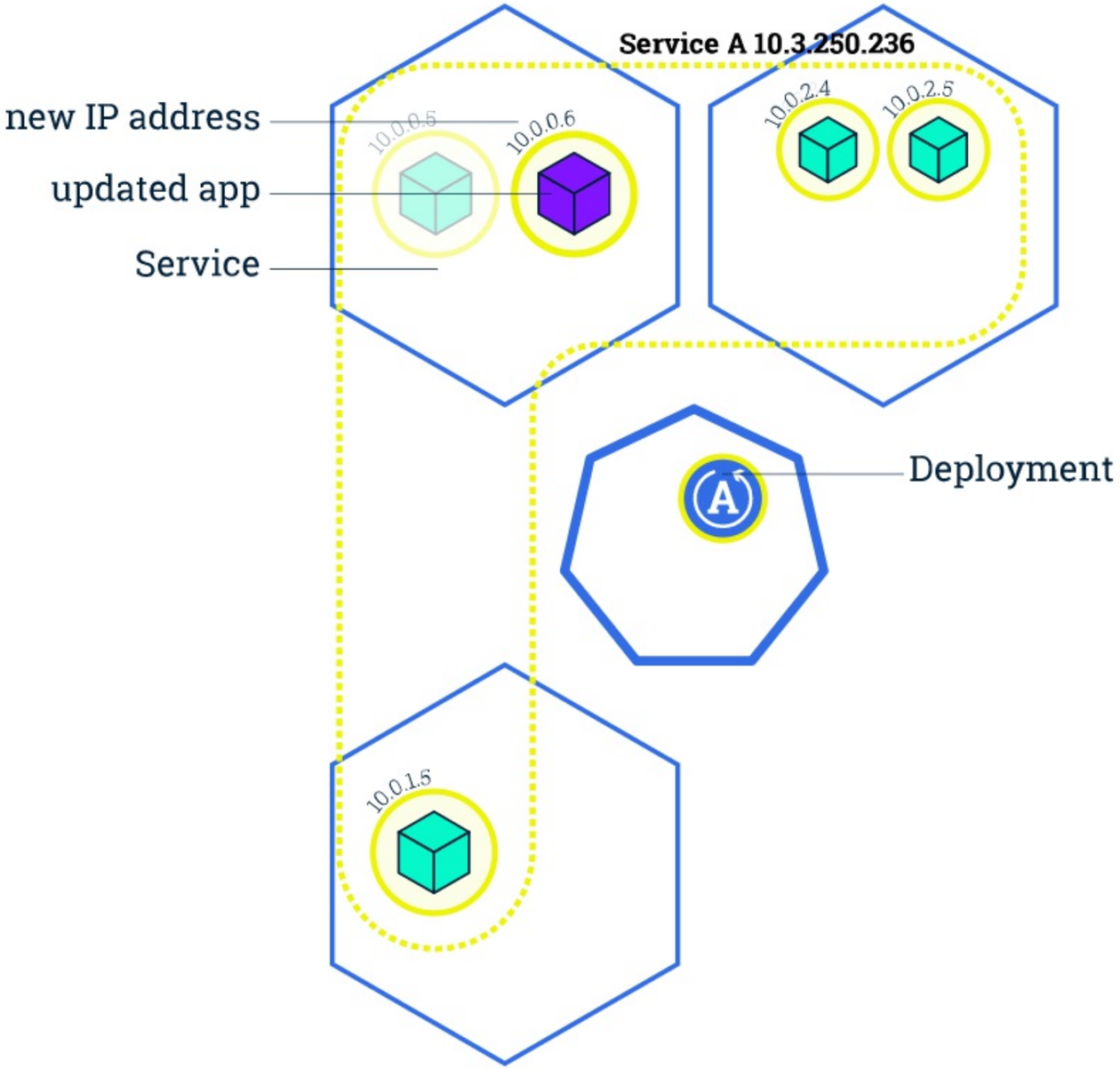

Service A 10.3.250.236
10.0.0.5
10.0.2.4
10.0.2.5
Service
Deployment
10.0.1.5

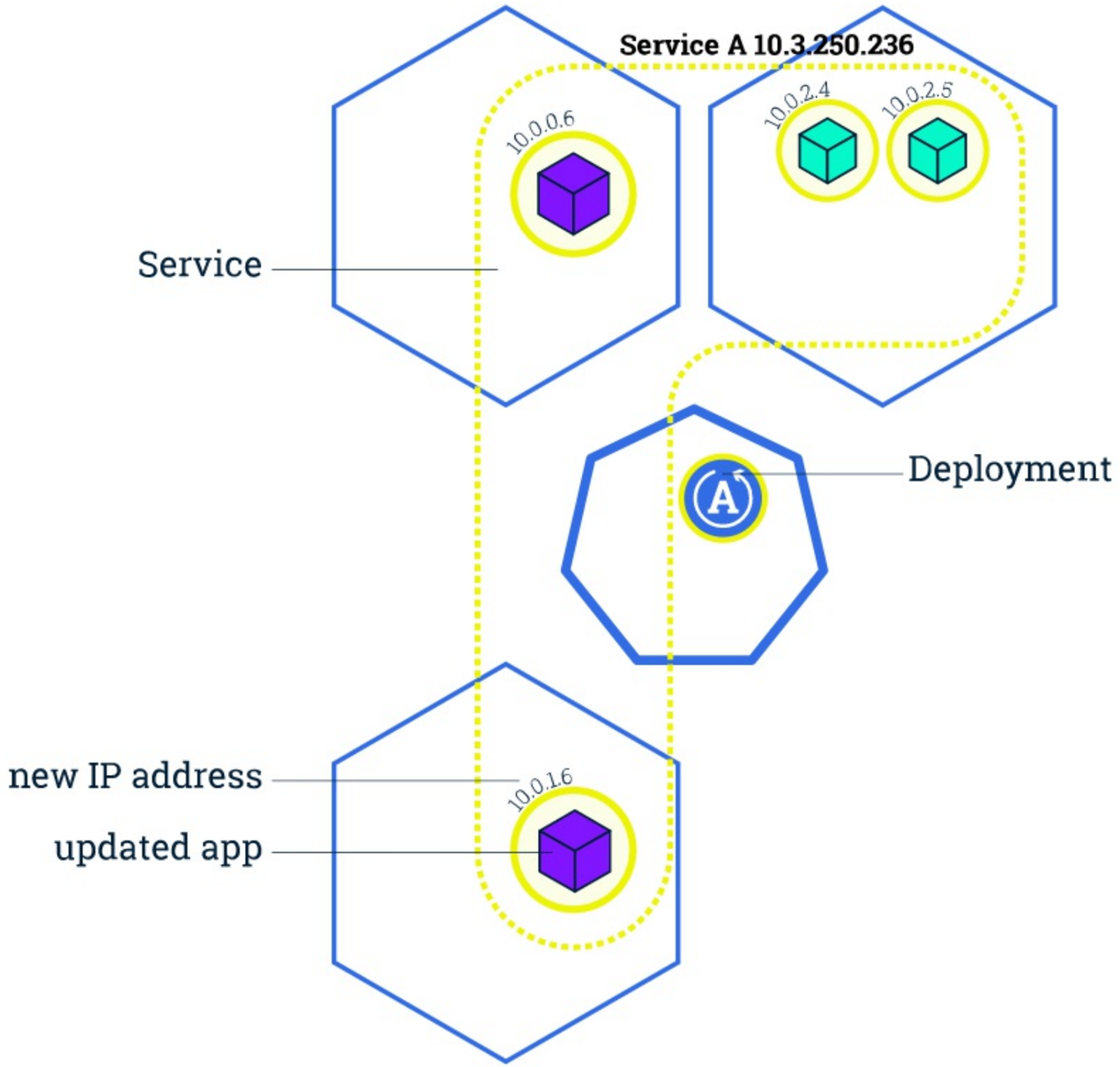
ROLLING UPDATES

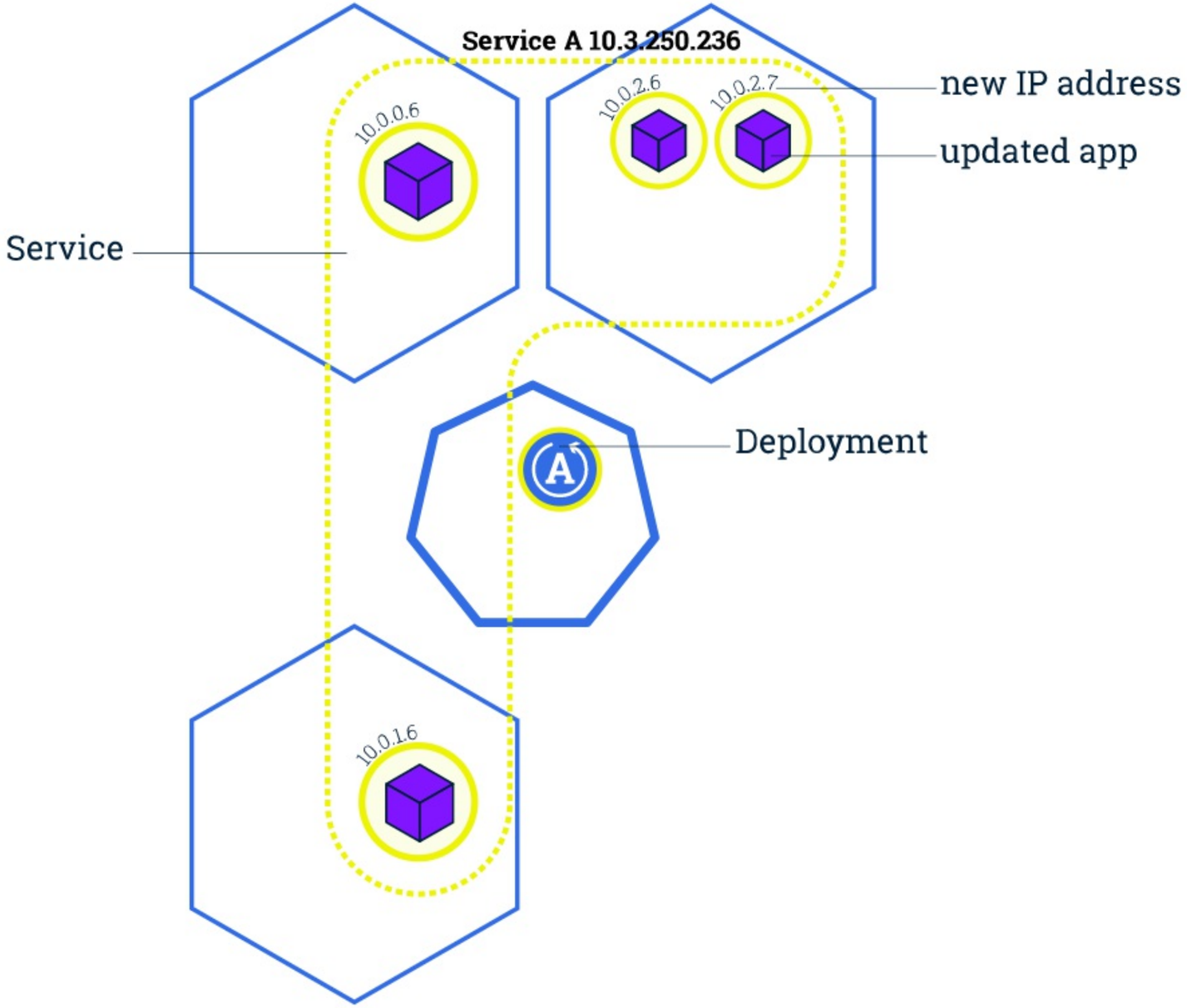
-

INCREMENTAL PODS UPDATE FOR DEPLOYMENTS

- CI/CD with zero downtime


Service A 10.3.250.236
new IP address
updated app
Service
10.0.0.5
10.0.0.6
10.0.2.4
10.0.2.5
Deployment
10.0.1.5


Service A 10.3.250.236
10.0.0.6
10.0.2.4
10.0.2.5
Service
Deployment
new IP address
updated app
10.0.1.6


Service A 10.3.250.236
10.0.0.6
10.0.2.6
10.0.2.7
new IP address
updated app
Service
Deployment
10.0.1.6

HOW TO DEPLOY ON KUBERNETES (DEMO)

TAKE AWAYS

- Containers are the new atomic distribution/packaging unit
- Scale your app in line with the demand
- Make inteligent use of available resources

THANKS!

Questions?