УНИВЕРСИТЕТ ИТМО

Introduction to Kubernetes

Nikolay Butakov, Sergey Teryoshkin

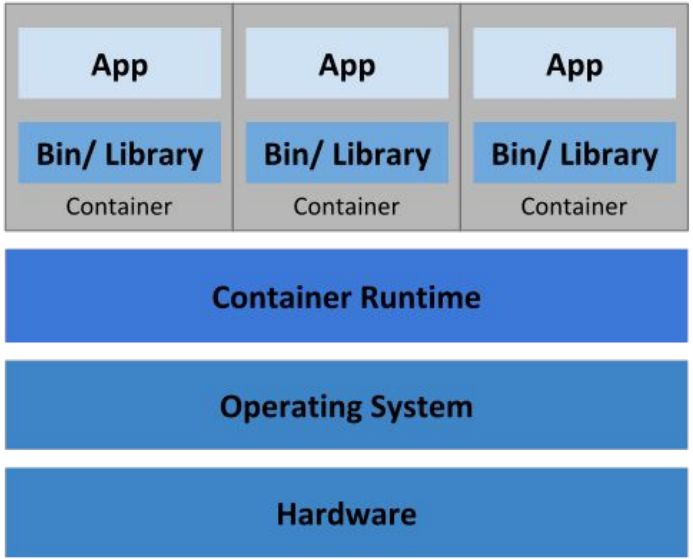
2022

Containers

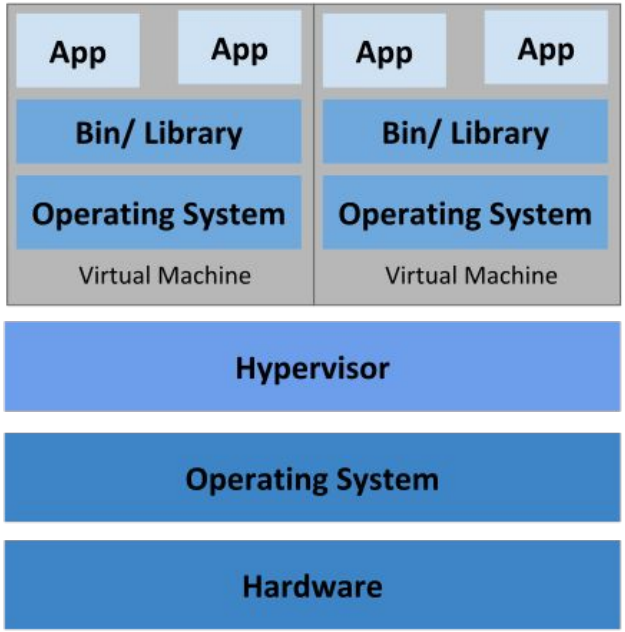
Containers are lightweight packages of your application code together with dependencies such as specific versions of programming language runtimes and libraries required to run your software services.

Containerization allows to share CPU, RAM, Storage, Network resources **at the operating system level**.

Containers



Container Deployment



Virtualized Deployment

https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/

Containers

ITMO UNIVERSITY

Containers mostly use several Linux Kernel features: cgroups and namespaces.

cgroups development was started at 2006 and originally had name “process containers”. In 2007 the name was changed to “**c**ontrol **groups**”.

namespaces originated in 2002.

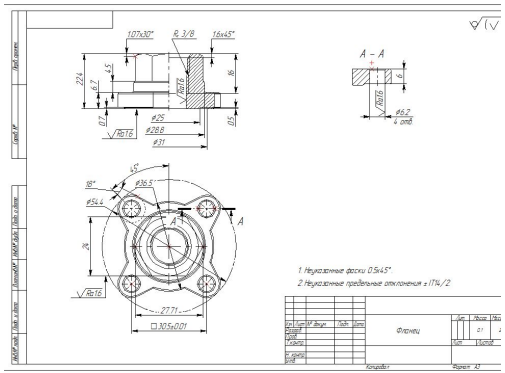
ITMO more than a UNIVERSITY

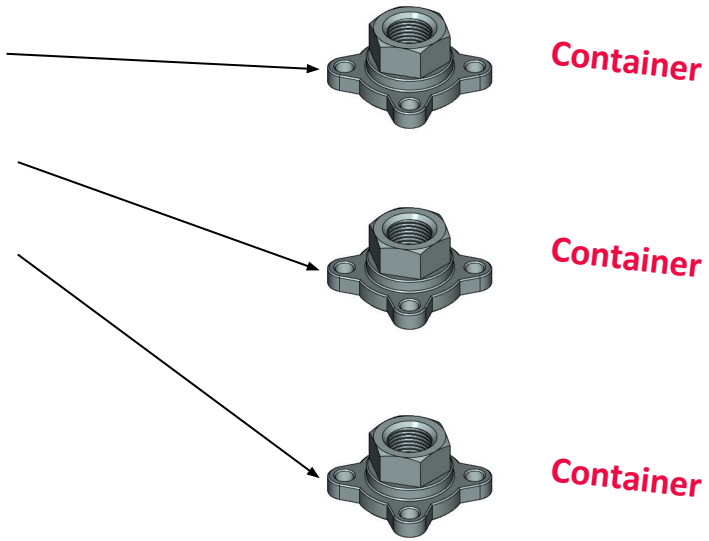
Docker

Docker is a containerization platform.

Docker allows to build, deploy and manage containers and ***images***.

Image





Docker

Open Container Initiative (**OCI**) **Image Specification** - standard that guarantees compatibility between images builded by different tools.

Alternatives:

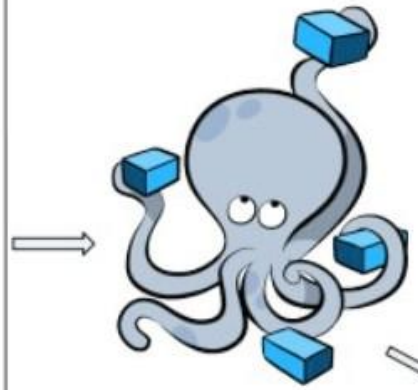
- Podman
- OpenVZ
- RKT
- LXD
- … others

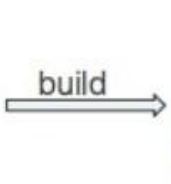
Docker-Compose

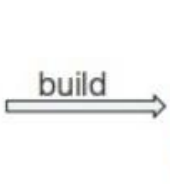
ITMO UNIVERSITY

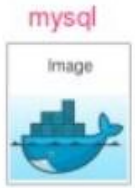
docker-compose.yaml

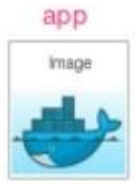
```
version: "3.7"
services:
 db:
  image: mysql:8.0.19
  restart: always
  environment:
   - MYSQL_DATABASE=example
   - MYSQL_ROOT_PASSWORD=password
 app:
  build: app
  restart: always
 web:
  build: web
  restart: always
  ports:
   - 80:80
```

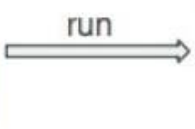


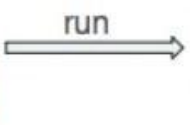


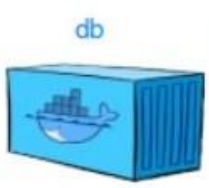












ITMO more than a UNIVERSITY

https://azuremarketplace.microsoft.com/ru-ru/marketplace/apps/cloud-infrastructure-services.docker-compose-2016

Container orchestration solutions

ITMO UNIVERSITY

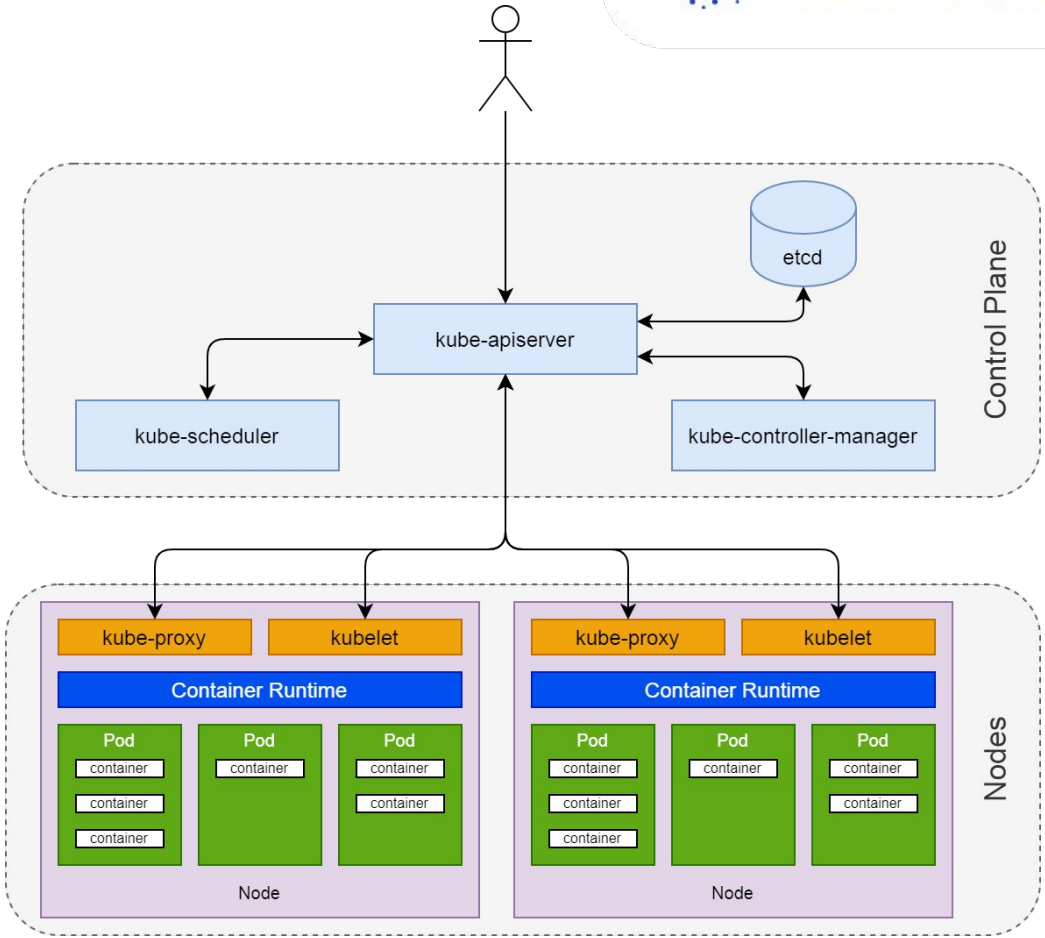
- Docker Swarm
- Nomad Hashicorp
- Kubernetes

ITMO more than a UNIVERSITY

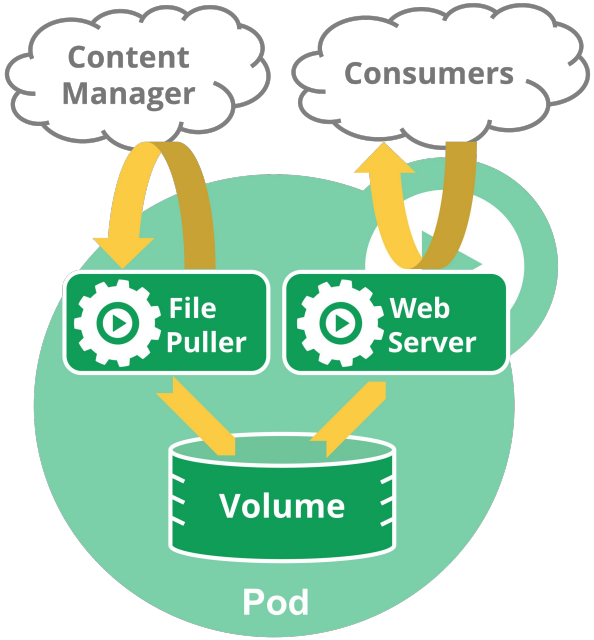
Kubernetes architecture overview

Main components:

- ApiServer
- etcd
- Scheduler
- Kube-Controller
- Kubelet
- Kube-Proxy



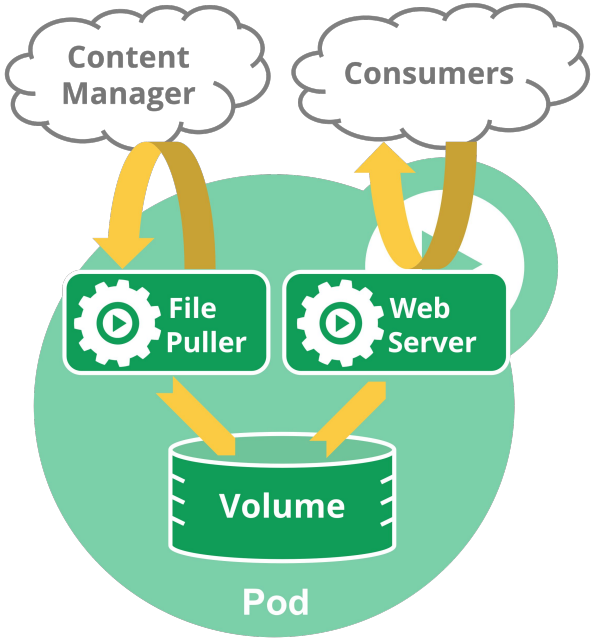
Pod



Pod - the smallest deployable Kubernetes unit.
Pod consists of one or more containers.
All pod's containers are located on the same machine.

https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/

Pod

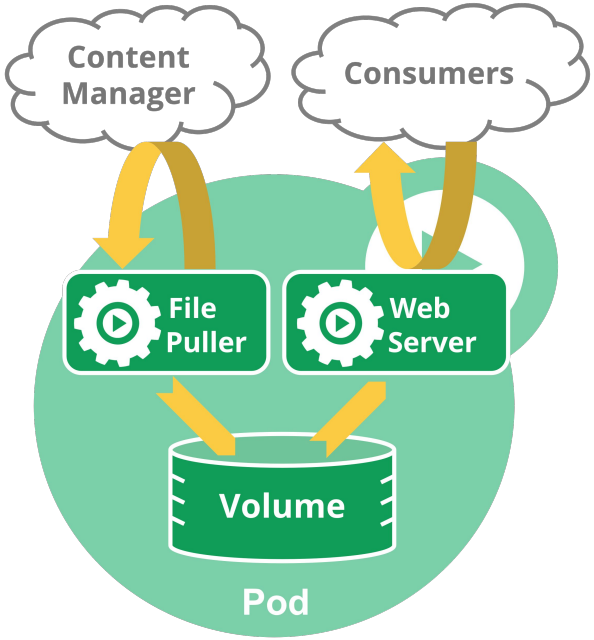


Pod - the <u>smallest</u> deployable Kubernetes unit.
Pod consists of one or more containers.
All pod's containers are located on the <u>same</u> machine.

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.14.2
```

https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/

Pod

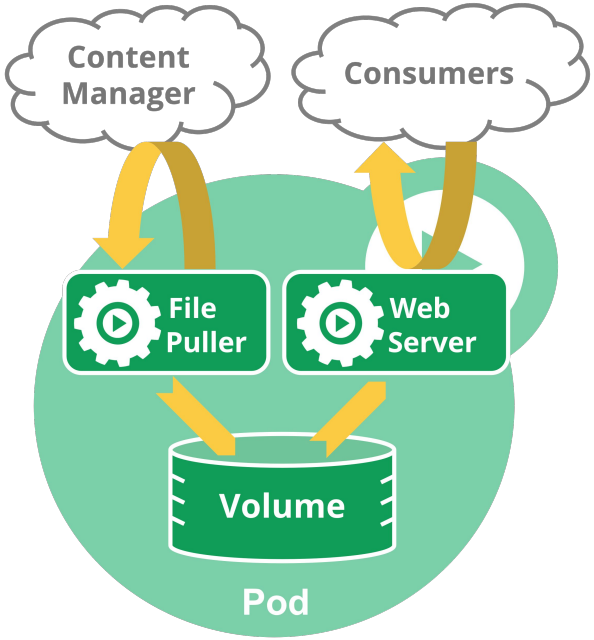


Pod - the <u>smallest</u> deployable Kubernetes unit.
Pod consists of one or more containers.
All pod's containers are located on the <u>same</u> machine.

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.14.2
```

https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/

Pod

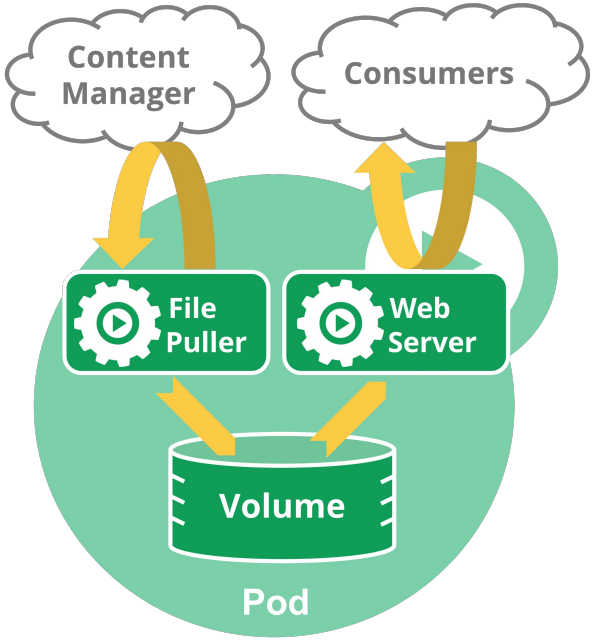


Pod - the <u>smallest</u> deployable Kubernetes unit.
Pod consists of one or more containers.
All pod's containers are located on the <u>same</u> machine.

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.14.2
```

https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/

Pod

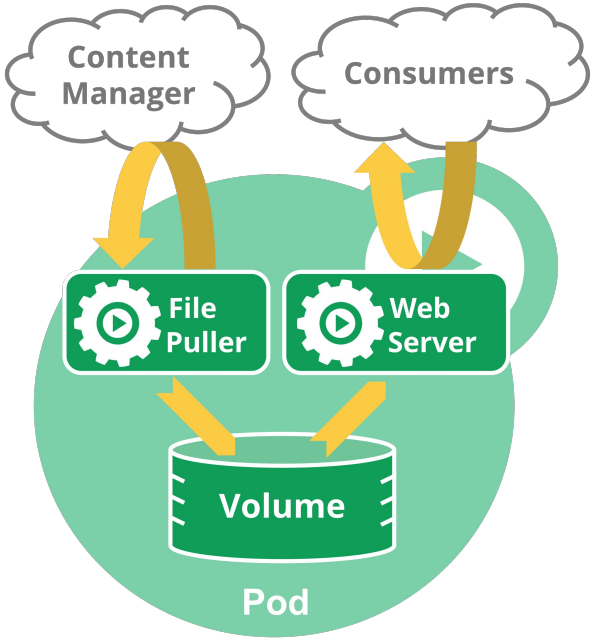


Pod - the smallest deployable Kubernetes unit.
Pod consists of one or more containers.
All pod's containers are located on the same machine.

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.14.2
```

https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/

Pod

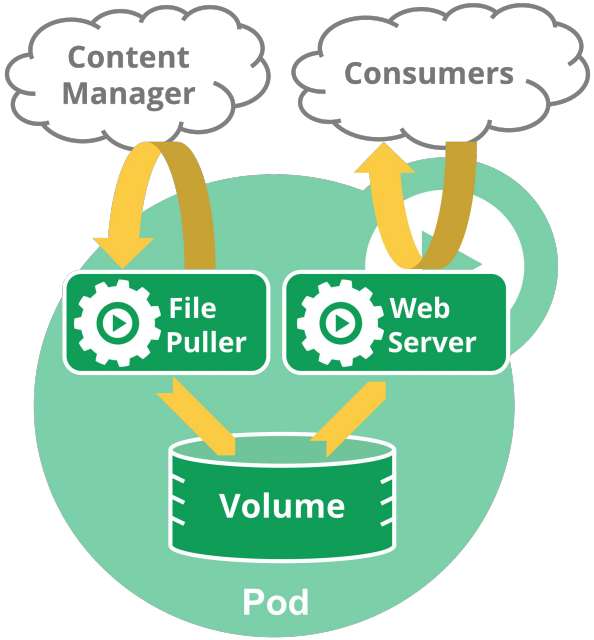


Pod - the smallest deployable Kubernetes unit.
Pod consists of one or more containers.
All pod's containers are located on the same machine.

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.14.2
```

https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/

Pod



Pod - the smallest deployable Kubernetes unit.
Pod consists of one or more containers.
All pod's containers are located on the same machine.

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.14.2
```

https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/

Kubectl

```
kubectl create -f nginx.yaml
pod/nginx created
```

Kubectl

```
kubectl create -f nginx.yaml
pod/nginx created

kubectl get pods
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   0          10s
```

Kubectl

```
kubectl create -f nginx.yaml
pod/nginx created

kubectl get pods
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   0          10s

kubectl describe pod nginx
/* <Detailed pod information> */
Events:
  Type    Reason     Age   From               Message
  ----    ------     ---   ----               -------
  Normal  Scheduled  2m    default-scheduler  Successfully assigned default/nginx to host-1
  Normal  Pulling    2m    kubelet            Pulling image "nginx:1.14.2"
  Normal  Pulled     2m    kubelet            Successfully pulled image "nginx:1.14.2" in 6.7s
  Normal  Created    2m    kubelet            Created container nginx
  Normal  Started    2m    kubelet            Started container nginx
```

Kubectl

ITMO UNIVERSITY

```
kubectl exec -it nginx -c nginx -- /bin/bash
```

ITMOre than a UNIVERSITY

Kubectl

```
kubectl exec -it nginx -c nginx -- /bin/bash
root@nginx:/#
```

Kubectl

ITMO UNIVERSITY

```
kubectl exec -it nginx -c nginx -- /bin/bash
root@nginx:/# ls -lah
total 76K
drwxr-xr-x   1 root root 4.0K Nov  3 21:14 .
drwxr-xr-x   1 root root 4.0K Nov  3 21:14 ..
-rwxr-xr-x   1 root root    0 Nov  3 21:14 .dockerenv
drwxr-xr-x   2 root root 4.0K Mar 26  2019 bin
drwxr-xr-x   2 root root 4.0K Feb  3  2019 boot
drwxr-xr-x   5 root root  360 Nov  3 21:14 dev
drwxr-xr-x   1 root root 4.0K Nov  3 21:14 etc
drwxr-xr-x   2 root root 4.0K Feb  3  2019 home
drwxr-xr-x   1 root root 4.0K Mar 26  2019 lib
drwxr-xr-x   2 root root 4.0K Mar 26  2019 lib64
drwxr-xr-x   2 root root 4.0K Mar 26  2019 media
drwxr-xr-x   2 root root 4.0K Mar 26  2019 mnt
drwxr-xr-x   2 root root 4.0K Mar 26  2019 opt
dr-xr-xr-x 454 root root    0 Nov  3 21:14 proc
drwx------   2 root root 4.0K Mar 26  2019 root
drwxr-xr-x   1 root root 4.0K Nov  3 21:14 run
drwxr-xr-x   2 root root 4.0K Mar 26  2019 sbin
drwxr-xr-x   2 root root 4.0K Mar 26  2019 srv
dr-xr-xr-x  13 root root    0 Nov  3 23:08 sys
drwxrwxrwt   1 root root 4.0K Mar 26  2019 tmp
drwxr-xr-x   1 root root 4.0K Mar 26  2019 usr
drwxr-xr-x   1 root root 4.0K Mar 26  2019 var
```

ITMO more than a UNIVERSITY

Pod

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image:
nginx:1.14.2
```



```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.15
```

ITMO more than a UNIVERSITY

Kubectl

```
kubectl create -f nginx.yaml
Error from server (AlreadyExists): error when creating "nginx.yaml": pods "nginx" already exists
```

Kubectl

ITMO UNIVERSITY

```
kubectl create -f nginx.yaml
Error from server (AlreadyExists): error when creating "nginx.yaml": pods "nginx" already exists

kubectl delete pod nginx
pod "nginx" deleted

kubectl create -f nginx.yaml
pod/nginx created
```

ITMO more than a UNIVERSITY

Kubectl

```
kubectl create -f nginx.yaml
Error from server (AlreadyExists): error when creating "nginx.yaml": pods "nginx" already exists

kubectl delete pod nginx
pod "nginx" deleted

kubectl create -f nginx.yaml
pod/nginx created

kubectl describe pod nginx
/* <Some detailed information> */
Containers:
  nginx:
    Container ID:   docker://<container-hash>
    Image:          nginx:1.15
    Image ID:       docker-pullable://nginx@sha256:<image-hash>
    Port:           <none>
```

Kubectl

ITMO UNIVERSITY

```
kubectl apply -f nginx.yaml
pod/nginx configured
```

ITMOre than a UNIVERSITY

Kubectl

ITMO UNIVERSITY

```
kubectl apply -f nginx.yaml
pod/nginx configured

kubectl describe pod nginx
/* <Some detailed information> */
Containers:
  nginx:
    Container ID:   docker://<container-hash>
    Image:          nginx:1.17
    Image ID:       docker-pullable://nginx@sha256:<image-hash>
    Port:           <none>
/* <Some detailed information> */
Events:
  Type    Reason     Age   From               Message
  ----    ------     ----  ----               -------
  Normal  Scheduled  24m   default-scheduler  Successfully assigned default/nginx to host-1
  Normal  Pulling    24m   kubelet            Pulling image "nginx:1.15"
  Normal  Pulled     24m   kubelet            Successfully pulled image "nginx:1.15" in 6.4s
  Normal  Killing    28s   kubelet            Container nginx definition changed, will be
restarted
  Normal  Pulling    28s   kubelet            Pulling image "nginx:1.17"
  Normal  Created    20s   kubelet            Created container nginx
  Normal  Pulled     20s   kubelet            Successfully pulled image "nginx:1.17" in 8.1s
  Normal  Started    19s   kubelet            Started container nginx
```

ITMO more than a UNIVERSITY

Namespace

Namespace - the way in Kubernetes to divide working areas by project, teams, departments, etc.

```
main-directory              (cluster)
├── awesome-project         (namespace)
│   ├── data.csv
│   ├── nginx.config
│   ├── nginx.pod
│   └── nginx.service
├── cute-team               (namespace)
│   ├── abrakadabra.wtf
│   ├── ivanov-data.json
│   └── sergeev-conf.cfg
├── exciting-project        (namespace)
│   ├── abcd
│   ├── conf.yaml
│   └── test-test
└── service-dir             (namespace)
    ├── 0eacfb5569bcc
    └── _tmp
```

Namespace

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: test
```

ITMO more than a UNIVERSITY

Namespace

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: test
```

```
kubectl get namespaces
NAME              STATUS   AGE
default           Active   47h
kube-node-lease   Active   47h
kube-public       Active   47h
kube-system       Active   47h
test              Active   80s
```

Namespace

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: test
```

```
kubectl get namespaces
NAME              STATUS   AGE
default           Active   47h
kube-node-lease   Active   47h
kube-public       Active   47h
kube-system       Active   47h
test              Active   80s
```

```
kubectl get pods -n test
No resources found in test namespace.
```

Namespace

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: test
```

```
kubectl get namespaces
NAME              STATUS   AGE
default           Active   47h
kube-node-lease   Active   47h
kube-public       Active   47h
kube-system       Active   47h
test              Active   80s
```

```
kubectl get pods -n test
No resources found in test namespace.
```

```
kubectl get pods -n kube-system
NAME                              READY   STATUS    RESTARTS   AGE
coredns-f9fd979d6-6cfzd           1/1     Running   0          2d
etcd-host-1                       1/1     Running   0          2d
kube-apiserver-host-1             1/1     Running   0          2d
kube-controller-manager-host-1    1/1     Running   0          2d
kube-proxy-kkhmv                  1/1     Running   0          2d
kube-scheduler-host-1             1/1     Running   0          2d
storage-provisioner               1/1     Running   0          2d
```

Namespace

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  namespace: test
spec:
  containers:
  - name: nginx
    image: nginx:1.17
```

Namespace

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  namespace: test
spec:
  containers:
  - name: nginx
    image: nginx:1.17
```

```
kubectl apply -f nginx.yaml -n test
pod/nginx created
```

Namespace

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  namespace: test
spec:
  containers:
  - name: nginx
    image: nginx:1.17
```

```
kubectl apply -f nginx.yaml -n test
pod/nginx created

kubectl get pods -n test
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   0          39s
```

Namespace

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  namespace: test
spec:
  containers:
  - name: nginx
    image: nginx:1.17
```

```
kubectl apply -f nginx.yaml -n test
pod/nginx created

kubectl get pods -n test
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   0          39s

kubectl get pods -n default
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   1          112m
```

Labels

Labels - effective way to organize Kubernetes object identifying to logical groups.

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    key: value
    lecture: k8s
    app: nginx
    awesomeLabel: “true”
spec:
  containers:
  - name: nginx
    image: nginx:1.17
```

Labels

```
kubectl describe pod nginx
Name:           nginx
Namespace:      default
Priority:       0
Node:           host-1/<node-IP>
Start Time:     Thu, 04 Nov 2021 14:22:01 +0300
Labels:         app=nginx
                awesomeLabel=true
                key=value
                lecture=k8s
Annotations:    <none>
Status:         Running
IP:             172.17.0.15
/* <Some detailed information> */
```

Labels

```
kubectl describe pod nginx
Name:           nginx
Namespace:      default
Priority:       0
Node:           host-1/<node-IP>
Start Time:     Thu, 04 Nov 2021 14:22:01 +0300
Labels:         app=nginx
                awesomeLabel=true
                key=value
                lecture=k8s
Annotations:    <none>
Status:         Running
IP:             172.17.0.15
/* <Some detailed information> */

kubectl get pods --selector=awesomeLabel=true
NAME     READY    STATUS     RESTARTS    AGE
nginx    1/1      Running    1           3h17m
```

Labels

```
kubectl describe pod nginx
Name:           nginx
Namespace:      default
Priority:       0
Node:           host-1/<node-IP>
Start Time:     Thu, 04 Nov 2021 14:22:01 +0300
Labels:         app=nginx
                awesomeLabel=true
                key=value
                lecture=k8s
Annotations:    <none>
Status:         Running
IP:             172.17.0.15
/* <Some detailed information> */

kubectl get pods --selector=awesomeLabel=true
NAME     READY    STATUS     RESTARTS    AGE
nginx    1/1      Running    1           3h17m

kubectl get pods -l awesomeLabel=true --all-namespaces
NAMESPACE    NAME     READY    STATUS     RESTARTS    AGE
default      nginx    1/1      Running    1           3h20m
test         nginx    1/1      Running    0           92m
```

Labels

NOT EQUAL

```
kubectl get pods -l awesomeLabel !=true
```

NOT IN

```
kubectl get pods -l 'awesomeLabel  notin (false, nottrue, no)'
```

IN

```
kubectl get pods -l 'awesomeLabel  in (true, yes)'
```

AND

```
kubectl get pods -l 'awesomeLabel  in (true, yes) ,lecture=k8s'
```

Environment variables

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.17
    env:
    - name: VAR1
      value: "12345"
    - name: VAR2
      value: wow!
```

ITMO more than a UNIVERSITY

Environment variables

```
kubectl describe pod nginx
/* <Some detailed information> *
    Ready:          True
    Restart Count: 0
    Environment:
      VAR1:  12345
      VAR2:  wow!
/* <Some detailed information> *
```

```
kubectl exec nginx -- env
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bi
HOSTNAME=nginx
VAR1=12345
VAR2=wow!
KUBERNETES_PORT_443_TCP=tcp://10.96.0.1:44
KUBERNETES_PORT_443_TCP_PROTO=tc
KUBERNETES_PORT_443_TCP_PORT=44
KUBERNETES_PORT_443_TCP_ADDR=10.96.0.
KUBERNETES_SERVICE_HOST=10.96.0.
KUBERNETES_SERVICE_PORT=44
KUBERNETES_SERVICE_PORT_HTTPS=44
KUBERNETES_PORT=tcp://10.96.0.1:44
NGINX_VERSION=1.17.1
NJS_VERSION=0.3.
PKG_RELEASE=1~buste
HOME=/root
```

Environment variables

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: first-service
spec:
  containers:
  - name: service
    image: service:latest
    env:
    - name: DB HOST
      value: postgres-host
    - name: DB PORT
      value: “5432”
```

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: second-service
spec:
  containers:
  - name: service
    image: another-service:latest
    env:
    - name: DB HOST
      value: postgres-host
    - name: DB PORT
      value: “5432”
```

Environment variables

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: first-service
spec:
  containers:
  - name: service
    image: service:latest
    env:
    - name: DB_HOST
      value: postgres-host
    - name: DB_PORT
      value: "5432"
```

NOT SCALABLE

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: second-service
spec:
  containers:
  - name: service
    image: another-service:latest
    env:
    - name: DB_HOST
      value: postgres-host
    - name: DB_PORT
      value: "5432"
```

MAINTENANCE COMPLEXITY

ITMO more than a UNIVERSITY

ConfigMap

ITMO UNIVERSITY

ConfigMap - an object to store the data in key-value pairs.

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: database-address
data:
  DB_HOST: postgres-host
  DB_PORT: “5432”
```

ITMO more than a UNIVERSITY

ConfigMap

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.17
    envFrom:
    - configMapRef:
        name: database-address
    env:
    - name: VAR1
      valueFrom:
        configMapKeyRef:
          name: database-address
          key: DB_PORT
    - name: VAR2
      value: wow!
```

ITMO more than a UNIVERSITY

ConfigMap

```
kubectl describe pod nginx
/* <Some detailed information> *
  Ready:          True
  Restart Count:  0
  Environment Variables from
    database-address  ConfigMap  Optional: false
  Environment:
    VAR1:  <set to the key 'DB_PORT' of config map 'database-address'>  Optional: false
    VAR2:  wow!
/* <Some detailed information> *
```

```
kubectl describe cm database-address
Name:         database-address
Namespace:    default
Labels:       <none>
Annotations:  <none>

Data
====
DB_PORT:
----
5432
DB_HOST:
----
postgres-host
Events:  <none>
```

ConfigMap

ITMO UNIVERSITY

```
kubectl exec nginx -- env
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/b
HOSTNAME=nginx
DB_HOST=postgres-host
DB_PORT=5432
VAR1=5432
VAR2=wow!
KUBERNETES_PORT_443_TCP=tcp://10.96.0.1:44
KUBERNETES_PORT_443_TCP_PROTO=tc
KUBERNETES_PORT_443_TCP_PORT=44
KUBERNETES_PORT_443_TCP_ADDR=10.96.0.
KUBERNETES_SERVICE_HOST=10.96.0.
KUBERNETES_SERVICE_PORT=44
KUBERNETES_SERVICE_PORT_HTTPS=44
KUBERNETES_PORT=tcp://10.96.0.1:44
NGINX_VERSION=1.17.1
NJS_VERSION=0.3.
PKG_RELEASE=1~buste
HOME=/root
```

ITMO more than a UNIVERSITY

Resources

ITMO UNIVERSITY

- Prevent negative influence due to unexpected behaviour
- Efficient resources utilization
- Scheduling is better

ITMO more than a UNIVERSITY

Resources

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.17
    resources:
      requests:
        memory: "256Mi"
        cpu: "0.5"
      limits:
        memory: "512Mi"
        cpu: "1"
```

ITMO more than a UNIVERSITY

Quality of Service (QoS)

ITMO UNIVERSITY

Quality of Service (QoS) class determines the pod's scheduling and eviction priority. QoS class is used by the Kubernetes scheduler to make decisions about scheduling pods onto nodes.

QoS classes:

- Guaranteed
- Burstable
- BestEffort

ITMO more than a UNIVERSITY

Guaranteed QoS

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.17
    resources:
      requests:
        memory: "512Mi"
        cpu: "1.5"
      limits:
        memory: "512Mi"
        cpu: "1.5"
```

Requests and Limits are the same
OR
No Requests, only Limits

ITMO more than a UNIVERSITY

Guaranteed QoS - Exclusive

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.17
    resources:
      requests:
        memory: "512Mi"
        cpu: "2"
      limits:
        memory: "512Mi"
        cpu: "2"
```

Guaranteed QoS + integer CPUs

- Sensitive to CPU throttling effects.
- Sensitive to context switches.
- Sensitive to processor cache misses.
- Benefits from sharing a processor resources (e.g., data and instruction caches).
- Sensitive to cross-socket memory traffic.
- Sensitive or requires hyperthreads from the same physical CPU core

ITMO more than a UNIVERSITY

Burstable QoS

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.17
    resources:
      requests:
        memory: "256Mi"
        cpu: "1"
      limits:
        memory: "512Mi"
        cpu: "2"
```

Requests and limits are specified and they are **different**.

OR

There are no limits specified.

ITMO more than a UNIVERSITY

BestEffort QoS

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.17
```

Requests and limits are not specified

“I do not care if my application receives enough resources”

Bad choice for CPU intensive application

ITMO more than a UNIVERSITY

Multi container Pod QoS

ITMO UNIVERSITY

- All the containers are Guaranteed => Guaranteed
- All the containers are BestEffort => BestEffort
- Otherwise - Burstable

ITMO more than a UNIVERSITY

Pods and Volumes

ITMO UNIVERSITY

How can I provide some data into my pods?

How can the pods save their data?

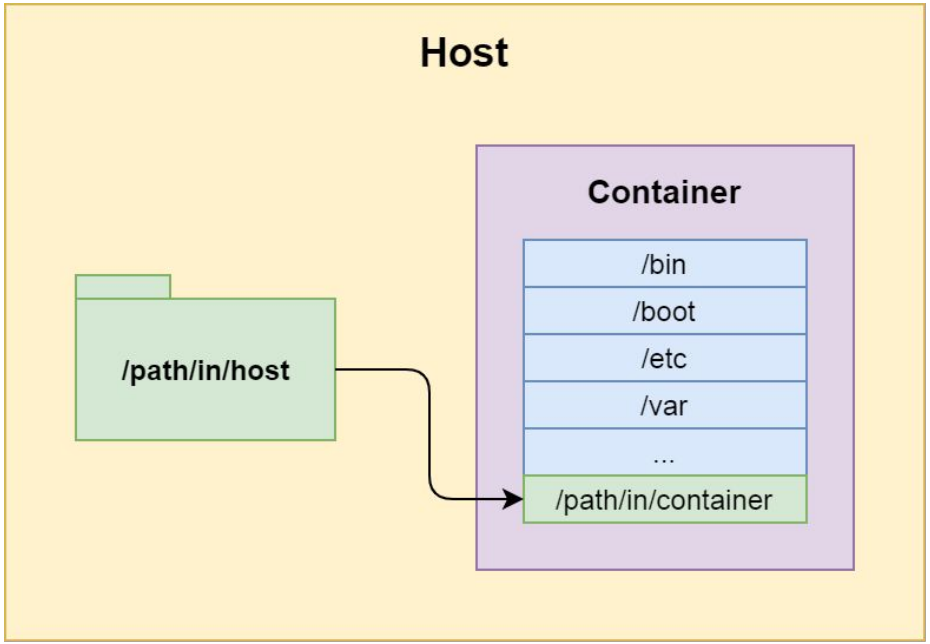
How does it work in case of multiple nodes?

What will happen with data after the pod deletion?

ITMO more than a UNIVERSITY

Docker volume mounting

```
docker run name \
  -v /path/in/host:/path/in/container \
  image:tag
```



HostPath

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  volumes:
  - name: awesome-data
    hostPath:
      path: /path/in/host
      type: Directory
  containers:
  - name: nginx
    image: nginx:1.17
    volumeMounts:
    - mountPath: /path/in/container
      name: awesome-data
      readOnly: false
```

Types:

- Directory
- File
- Socket
- FileOrCreate
- DirectoryOrCreate
- CharDevice
- BlockDevice

HostPath

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  volumes:
  - name: awesome-data
    hostPath:
      path: /path/in/host
      type: Directory
  containers:
  - name: nginx
    image: nginx:1.17
    volumeMounts:
    - mountPath: /path/in/container
      name: awesome-data
      readOnly: false
```

Types:

- Directory
- File
- Socket
- FileOrCreate
- DirectoryOrCreate
- CharDevice
- BlockDevice

```
kubectl exec -it nginx -c nginx -- /bin/bash
root@nginx:/# ls -lah /path/in/container/
total 8.0K
drwxrwxr-x 2 1000 1000 4.0K Nov  5 13:15 .
drwxr-xr-x 3 root root 4.0K Nov  5 13:17 ..
-rw-rw-r-- 1 1000 1000    0 Nov  5 13:15 hello.file
```

ITMO more than a UNIVERSITY

HostPath

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  volumes:
  - name: awesome-data
    hostPath:
      path: /path/in/host
      type: Directory
  containers:
  - name: nginx
    image: nginx:1.17
    volumeMounts:
    - mountPath: /path/in/container
      name: awesome-data
      readOnly: false
```

Types:

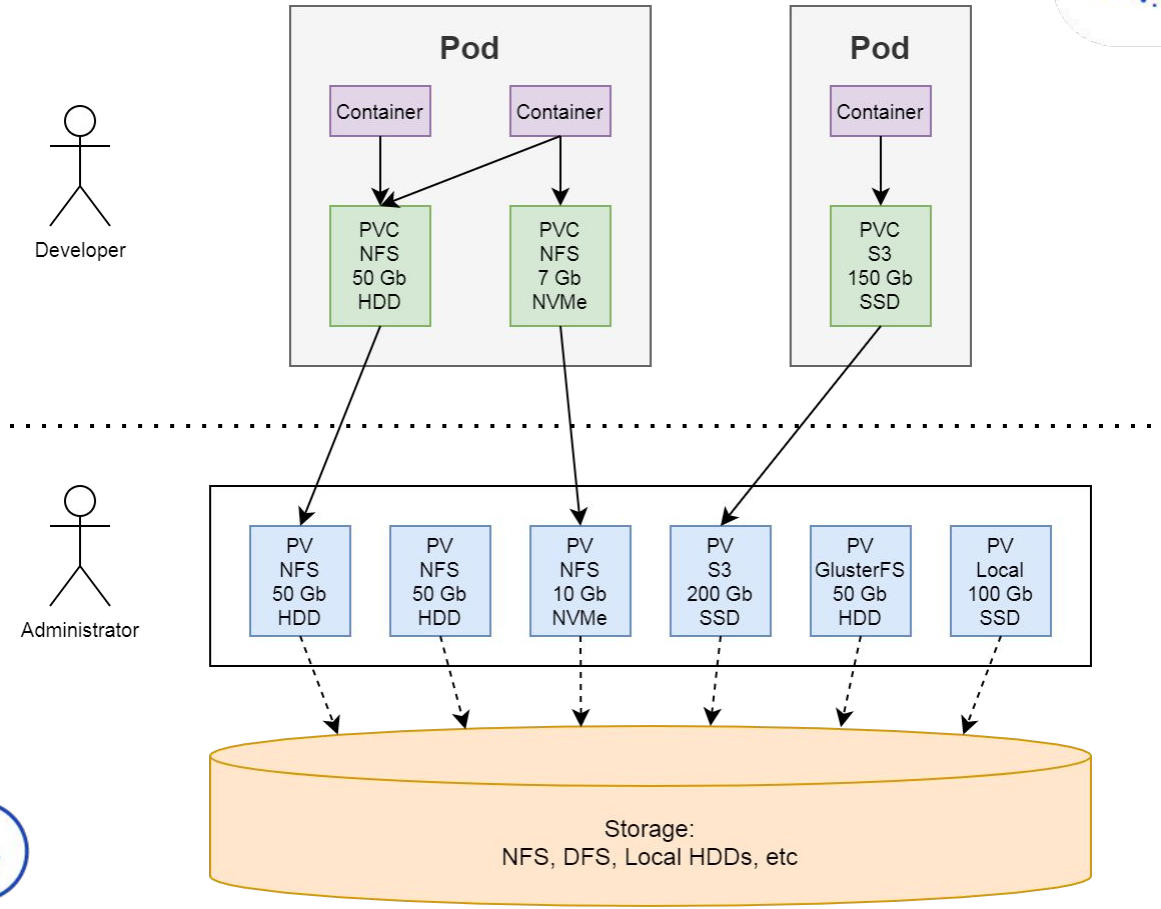
- Directory
- File
- Socket
- FileOrCreate
- DirectoryOrCreate
- CharDevice
- BlockDevice

```
kubectl exec -it nginx -c nginx -- /bin/bash
root@nginx:/# ls -lah /path/in/container/
total 8.0K
drwxrwxr-x 2 1000 1000 4.0K Nov  5 13:15 .
drwxr-xr-x 3 root root 4.0K Nov  5 13:17 ..
-rw-rw-r-- 1 1000 1000    0 Nov  5 13:15 hello.file
```

- Insecure
- Data is located only on one machine
- Unexpected behaviour
- Permission issues

ITMO more than a UNIVERSITY

PersistentVolume & PersistentVolumeClaim



PersistentVolume Types

ITMO UNIVERSITY

There are a number types of PV:

- awsElasticBlockStore
- azureDisk
- cephfs
- csi
- gcePersistentDisk
- glusterfs
- local
- nfs
- … others

ITMOre than a UNIVERSITY

PersistentVolume Example

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

ITMO more than a UNIVERSITY

PersistentVolume

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

ITMO more than a UNIVERSITY

PersistentVolume

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

- Filesystem (default)
- Block

PersistentVolume

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

A volume can only be mounted using **one access mode at a time**

Available modes:

- ReadWriteMany (RWX)
- ReadWriteOnce (RWO)
- ReadWriteOncePod (RWOP)
- ReadOnlyMany (ROX)

PersistentVolume

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

Policies:

- Retain
- Recycle (deprecated)
- Delete

PersistentVolume

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

ITMO more than a UNIVERSITY

PersistentVolume

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

ITMO more than a UNIVERSITY

PersistentVolume

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

ITMO more than a UNIVERSITY

PersistentVolume

ITMO UNIVERSITY

```
kubectl apply -f pv.yaml
persistentvolume/my-pv created

kubectl get pv -l lecture=k8s
NAME    CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM   STORAGECLASS   REASON   A
my-pv   5Gi        RWX            Retain           Available           nfs                     3m2
```

ITMO more than a UNIVERSITY

PersistentVolume

```
kubectl apply -f pv.yaml
persistentvolume/my-pv created

kubectl get pv -l lecture=k8s
NAME    CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM   STORAGECLASS   REASON   A
my-pv   5Gi        RWX            Retain           Available           nfs                     3m

kubectl describe pv my-pv
Name:            my-pv
Labels:          lecture=k8s
Annotations:     <none>
Finalizers:      [kubernetes.io/pv-protection]
StorageClass:    nfs
Status:          Available
Claim:
Reclaim Policy:  Retain
Access Modes:    RWX
VolumeMode:      Filesystem
Capacity:        5Gi
Node Affinity:   <none>
Message:
Source:
    Type:      NFS (an NFS mount that lasts the lifetime of a pod)
    Server:    127.0.0.1
    Path:      /data_300gb/nfs/lecture
    ReadOnly:  false
Events:        <none>
```

PersistentVolume

ITMO UNIVERSITY

```
kubectl apply -f pv.yaml
persistentvolume/my-pv configured

kubectl describe pv my-pv
Name:            my-pv
Labels:          lecture=k8s
Annotations:     <none>
Finalizers:      [kubernetes.io/pv-protection]
StorageClass:    super-nfs
Status:          Available
Claim:
Reclaim Policy:  Retain
Access Modes:    RWX
VolumeMode:      Filesystem
Capacity:        5Gi
Node Affinity:   <none>
Message:
Source:
    Type:      NFS (an NFS mount that lasts the lifetime of a pod)
    Server:    127.0.0.1
    Path:      /data_300gb/nfs/lectur
    ReadOnly:  false
Events:        <none>
```

ITMO More than a UNIVERSITY

PersistentVolumeClaim

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
  labels:
    lecture: k8s
spec:
  storageClassName: nfs
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 5Gi
  selector:
    matchLabels:
      lecture: k8s
  # volumeName: my-pv
```

ITMO more than a UNIVERSITY

PersistentVolumeClaim

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
  labels:
    lecture: k8s
spec:
  storageClassName: nfs
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 5Gi
  selector:
    matchLabels:
      lecture: k8s
  # volumeName: my-pv
```

ITMO more than a UNIVERSITY

PersistentVolumeClaim

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
  labels:
    lecture: k8s
spec:
  storageClassName: nfs
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 5Gi
  selector:
    matchLabels:
      lecture: k8s
  # volumeName: my-pv
```

ITMO more than a UNIVERSITY

PersistentVolumeClaim

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
  labels:
    lecture: k8s
spec:
  storageClassName: nfs
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 5Gi
  selector:
    matchLabels:
      lecture: k8s
  # volumeName: my-pv
```

ITMO more than a UNIVERSITY

PersistentVolumeClaim

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
  labels:
    lecture: k8s
spec:
  storageClassName: nfs
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 5Gi
  selector:
    matchLabels:
      lecture: k8s
  # volumeName: my-pv
```

ITMO more than a UNIVERSITY

PersistentVolumeClaim

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: my-pv
  labels:
    lecture: k8s
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: nfs
  mountOptions:
    - relatime
  nfs:
    path: /data
    readOnly: no
    server: 127.0.0.1
```

```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
  labels:
    lecture: k8s
spec:
  storageClassName: nfs
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 5Gi
  selector:
    matchLabels:
      lecture: k8s
  # volumeName: my-pv
```

ITMO more than a UNIVERSITY

PersistentVolumeClaim

```
kubectl apply -f pvc.yaml
persistentvolumeclaim/my-pvc created

kubectl get pvc
NAME      STATUS   VOLUME   CAPACITY   ACCESS MODES   STORAGECLASS   AGE
my-pvc    Bound    my-pv    5Gi        RWX            nfs            92s

kubectl get pv -l lecture=k8s
NAME    CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM             STORAGECLASS   REASON   A
my-pv   5Gi        RWX            Retain           Bound    default/my-pvc    nfs                     52
```

PersistentVolumeClaim

```
kubectl apply -f pvc.yaml
persistentvolumeclaim/my-pvc created

kubectl get pvc
NAME      STATUS    VOLUME    CAPACITY    ACCESS MODES    STORAGECLASS    AGE
my-pvc    Bound     my-pv     5Gi         RWX             nfs             92s

kubectl get pv -l lecture=k8s
NAME    CAPACITY    ACCESS MODES    RECLAIM POLICY    STATUS    CLAIM             STORAGECLASS    REASON    A
my-pv   5Gi         RWX             Retain            Bound     default/my-pvc    nfs                       52

kubectl describe pvc my-pvc
Name:          my-pvc
Namespace:     default
StorageClass:  nfs
Status:        Bound
Volume:        my-pv
Labels:        lecture=k8s
Annotations:   pv.kubernetes.io/bind-completed: ye
               pv.kubernetes.io/bound-by-controller: ye
Finalizers:    [kubernetes.io/pvc-protection
Capacity:      5Gi
Access Modes:  RWX
VolumeMode:    Filesyster
Mounted By:    <none>
Events:        <none>
```

PersistentVolumeClaim

ITMO UNIVERSITY

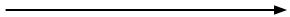
```
kubectl delete pv my-pv
persistentvolume "my-pv" deleted

kubectl get pv -l lecture=k8s
NAME    CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS        CLAIM             STORAGECLASS   REASON   A
my-pv   5Gi        RWX            Retain           Terminating   default/my-pvc    nfs                     66
```

ITMO more than a UNIVERSITY

Pod & PVC

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  volumes:
  - name: awesome-data
    hostPath:
      path: /path/in/host
      type: Directory
  containers:
  - name: nginx
    image: nginx:1.17
    volumeMounts:
    - mountPath: /path/in/container
      name: awesome-data
      readOnly: false
```



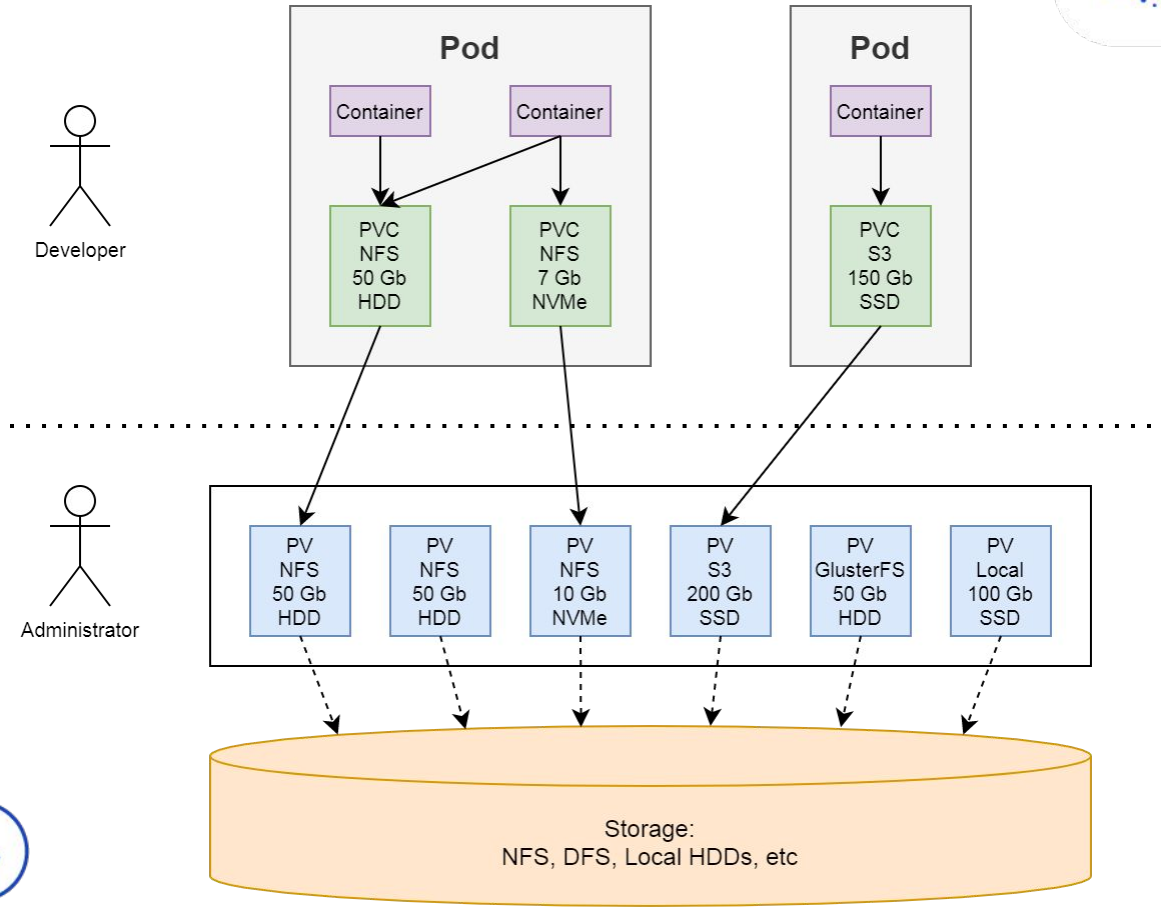
```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    lecture: k8s
    app: nginx
spec:
  volumes:
  - name: awesome-data
    persistentVolumeClaim:
      claimName: my-pvc
  containers:
  - name: nginx
    image: nginx:1.17
    volumeMounts:
    - mountPath: /path/in/container
      name: awesome-data
      readOnly: false
```

ITMO more than a UNIVERSITY

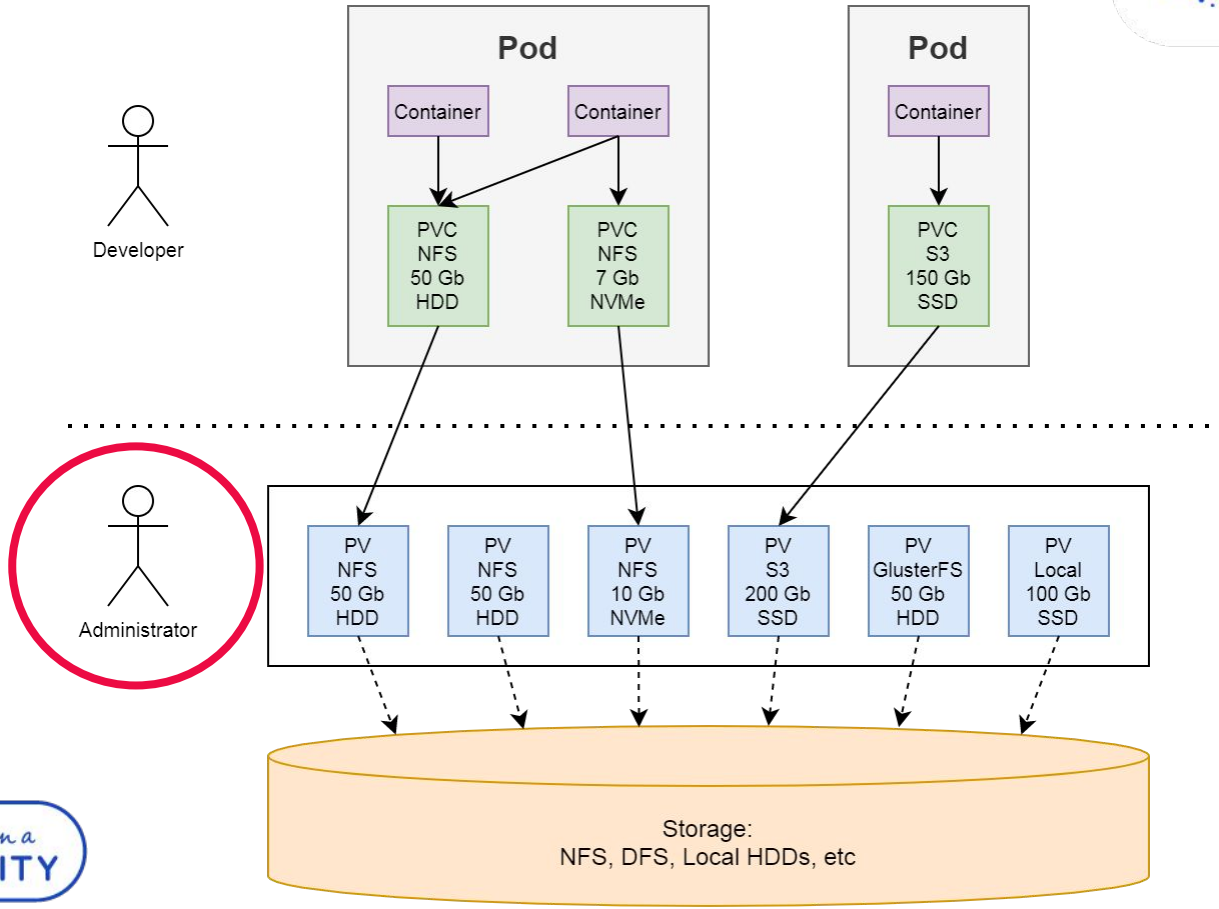
Pod & PVC

```
kubectl exec -it nginx -c nginx -- /bin/bash
root@nginx:/# ls -lah /path/in/container/
total 8.0K
drwxrwxr-x 2 1000 1000 4.0K Nov  8 14:43 .
drwxr-xr-x 3 root root 4.0K Nov  9 09:58 ..
root@nginx:/# touch /path/in/container/hello.file
root@nginx:/# ls -lah /path/in/container/
total 8.0K
drwxrwxr-x 2 1000 1000 4.0K Nov  9 09:58 .
drwxr-xr-x 3 root root 4.0K Nov  9 09:58 ..
-rw-r--r-- 1 root root    0 Nov  9 09:58 hello.file
root@nginx:/# exit
exit
admin@host-1:~/k8s-lecture$ ls -lah /data_300gb/nfs/lecture/
total 8,0K
drwxrwxr-x 2 admin admin 4,0K Nov  9 12:58 .
drwxrwxrwx 5 root  root  4,0K Nov  8 17:43 ..
-rw-r--r-- 1 root  root    0  Nov  9 12:58 hello.file
```

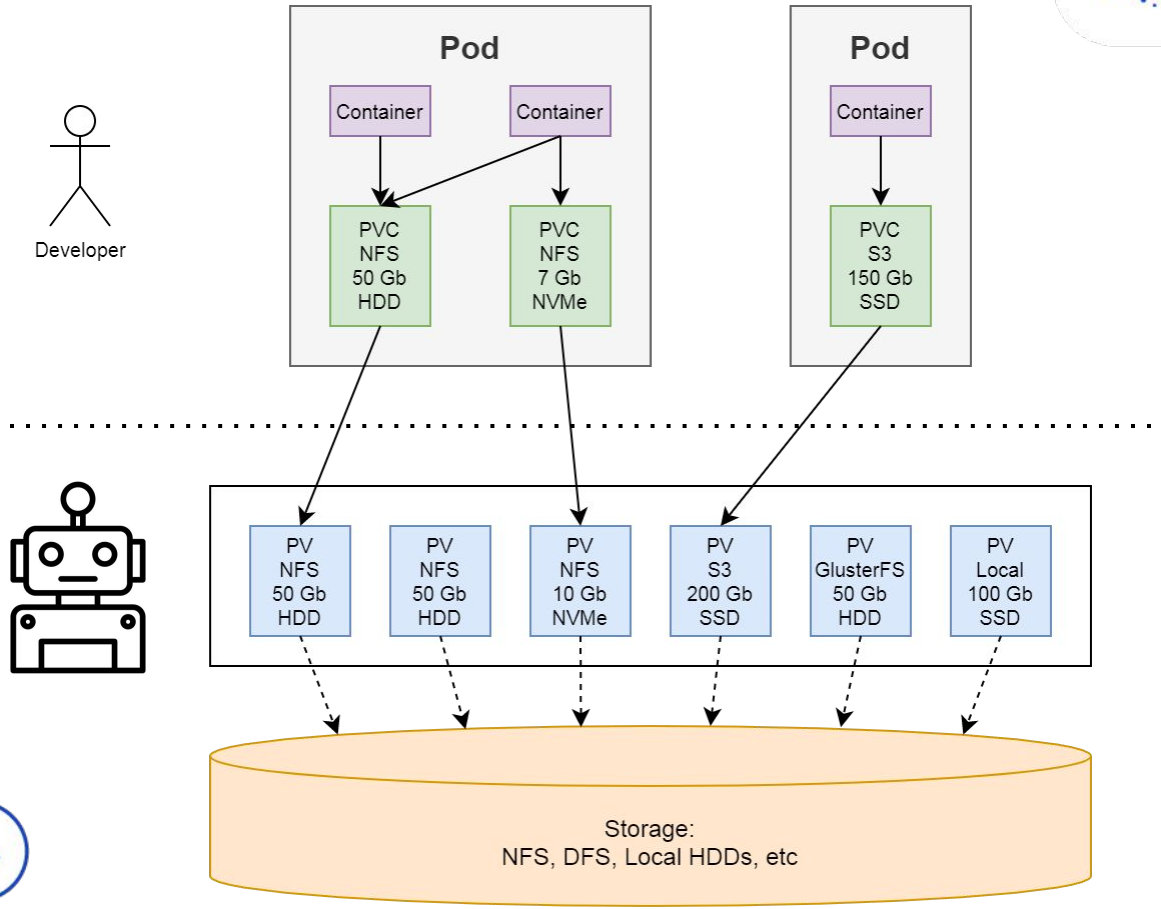
PersistentVolume & PersistentVolumeClaim



PersistentVolume & PersistentVolumeClaim



Volume Provisioner



Pods communication

```
kubectl get pods -o wide
NAME        READY   STATUS    RESTARTS   AGE    IP             NODE     NOMINATED NODE   READINESS GAT
net-tools   1/1     Running   0          32m    172.17.0.23    host-1   <none>           <nor
nginx       1/1     Running   0          100m   172.17.0.15    host-1   <none>           <nor
```

Pods communication

```
kubectl get pods -o wide
NAME        READY   STATUS    RESTARTS   AGE    IP             NODE     NOMINATED NODE   READINESS GAT
net-tools   1/1     Running   0          32m    172.17.0.23    host-1   <none>           <nor
nginx       1/1     Running   0          100m   172.17.0.15    host-1   <none>           <nor

kubectl exec -it net-tools -- bash
bash-5.1# ping 172.17.0.15
PING 172.17.0.15 (172.17.0.15) 56(84) bytes of data.
64 bytes from 172.17.0.15: icmp_seq=1 ttl=64 time=0.140 ms
64 bytes from 172.17.0.15: icmp_seq=2 ttl=64 time=0.055 ms
64 bytes from 172.17.0.15: icmp_seq=3 ttl=64 time=0.059 ms
^C
--- 172.17.0.15 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2026ms
rtt min/avg/max/mdev = 0.055/0.084/0.140/0.039 ms

bash-5.1# curl http://172.17.0.15:8080/
Server address: 172.17.0.24:8080
Server name: nginx
Date: 09/Nov/2021:12:23:26 +0000
URI: /
Request ID: c3a213e2509466bb0bbb284402fdaa51
```

Pods communication

ITMO UNIVERSITY

```
kubectl get pods -o wide
NAME        READY   STATUS    RESTARTS   AGE    IP             NODE     NOMINATED NODE   READINESS GAT
net-tools   1/1     Running   0          32m    172.17.0.23    host-1   <none>           <nor
nginx       1/1     Running   0          100m   172.17.0.15    host-1   <none>           <nor

kubectl exec -it net-tools -- bash
bash-5.1# ping 172.17.0.15
PING 172.17.0.15 (172.17.0.15) 56(84) bytes of data.
64 bytes from 172.17.0.15: icmp_seq=1 ttl=64 time=0.140 ms
64 bytes from 172.17.0.15: icmp_seq=2 ttl=64 time=0.055 ms
64 bytes from 172.17.0.15: icmp_seq=3 ttl=64 time=0.059 ms
^C
--- 172.17.0.15 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2026ms
rtt min/avg/max/mdev = 0.055/0.084/0.140/0.039 ms

bash-5.1# curl http://172.17.0.15:8080/
Server address: 172.17.0.24:8080
Server name: nginx
Date: 09/Nov/2021:12:23:26 +0000
URI: /
Request ID: c3a213e2509466bb0bbb284402fdaa51
```

ITMO more than a UNIVERSITY

```
kubectl logs nginx -c nginx
172.17.0.23 - - [09/Nov/2021:12:23:26 +0000] "GET / HTTP/1.1" 200 612 "-" "curl/7.79.1" "
```

Pods communication

```
kubectl get pods -o wide
NAME        READY   STATUS    RESTARTS   AGE     IP             NODE     NOMINATED NODE   READINESS GAT
net-tools   1/1     Running   0          32m     172.17.0.23    host-1   <none>           <nor
nginx       1/1     Running   0          15s     172.17.0.24    host-1   <none>           <nor
```

Pods communication

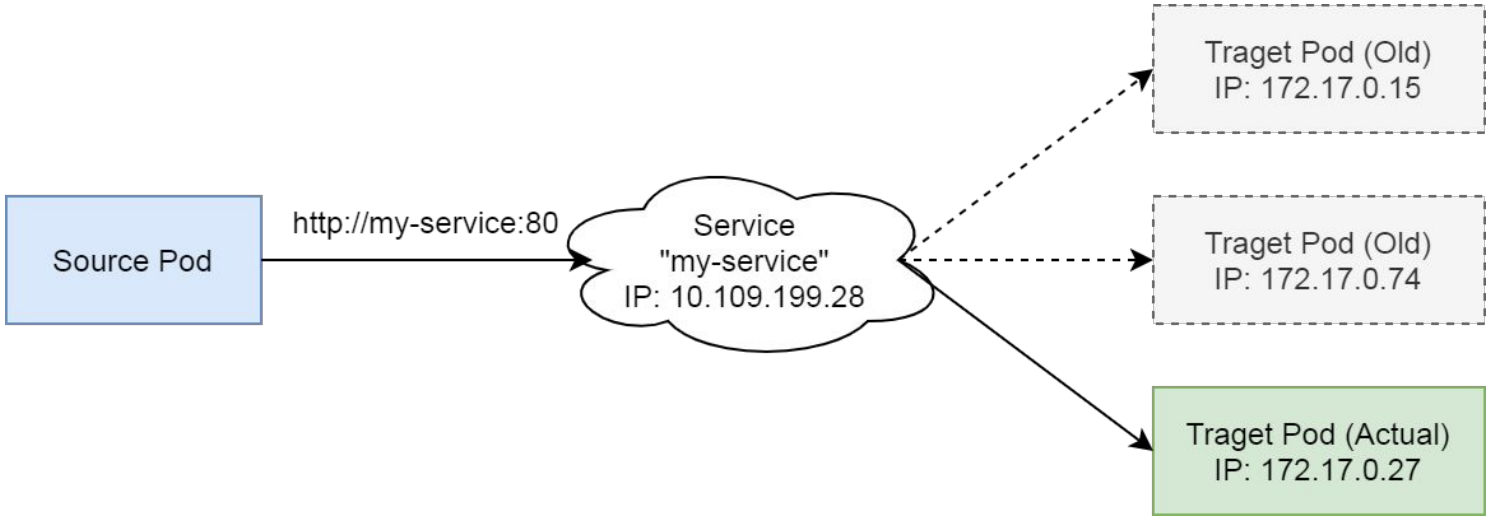
ITMO UNIVERSITY

```
kubectl get pods -o wide
NAME        READY   STATUS    RESTARTS   AGE    IP             NODE     NOMINATED NODE   READINESS GAT
net-tools   1/1     Running   0          32m    172.17.0.23    host-1   <none>           <nor
nginx       1/1     Running   0          15s    172.17.0.24    host-1   <none>           <nor
```

```
kubectl exec -it net-tools -- bash
bash-5.1# ping -i 1 172.17.0.15
PING 172.17.0.15 (172.17.0.15) 56(84) bytes of data.
From 172.17.0.23 icmp_seq=1 Destination Host Unreachable
From 172.17.0.23 icmp_seq=2 Destination Host Unreachable
From 172.17.0.23 icmp_seq=3 Destination Host Unreachable
^C
--- 172.17.0.15 ping statistics ---
6 packets transmitted, 0 received, +3 errors, 100% packet loss, time 5117ms

bash-5.1# ping 172.17.0.24
PING 172.17.0.24 (172.17.0.24) 56(84) bytes of data.
64 bytes from 172.17.0.24: icmp_seq=1 ttl=64 time=0.202 ms
64 bytes from 172.17.0.24: icmp_seq=2 ttl=64 time=0.065 ms
^C
--- 172.17.0.24 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1024ms
rtt min/avg/max/mdev = 0.065/0.133/0.202/0.068 ms
```

ITMO more than a UNIVERSITY

Service


Source Pod
http://my-service:80
Service
"my-service"
IP: 10.109.199.28
Traget Pod (Old)
IP: 172.17.0.15
Traget Pod (Old)
IP: 172.17.0.74
Traget Pod (Actual)
IP: 172.17.0.27

Service

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Service
metadata:
  name: my-service
  labels:
    lecture: k8s
spec:
  ports:
    - port: 80
      targetPort: 8080
  selector:
    app: nginx
    lecture: k8s
```

ITMO more than a UNIVERSITY

Service

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Service
metadata:
  name: my-service
  labels:
    lecture: k8s
spec:
  ports:
    - port: 80
      targetPort: 8080
  selector:
    app: nginx
    lecture: k8s
```

ITMO more than a UNIVERSITY

Service

```yaml
apiVersion: v1
kind: Service
metadata:
  name: my-service
  labels:
    lecture: k8s
spec:
  ports:
    - port: 80
      targetPort: 8080
  selector:
    app: nginx
    lecture: k8s
```

```
docker run --name \
  -p 80:8080 \
  image:tag
```

Service

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Service
metadata:
  name: my-service
  labels:
    lecture: k8s
spec:
  ports:
    - port: 80
      targetPort: 8080
  selector:
    app: nginx
    lecture: k8s
```

ITMO more than a UNIVERSITY

Service

```
kubectl apply -f service.yaml
service/my-service created

kubectl get services
NAME          TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)   AG
kubernetes    ClusterIP   10.96.0.1       <none>        443/TCP   7d
my-service    ClusterIP   10.109.199.28   <none>        80/TCP    2
```

Service

```
kubectl apply -f service.yaml
service/my-service create

kubectl get services
NAME         TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)   A
kubernetes   ClusterIP   10.96.0.1       <none>        443/TCP   7d
my-service   ClusterIP   10.109.199.28   <none>        80/TCP

kubectl describe service my-service
Name:              my-servic
Namespace:         defaul
Labels:            lecture=k8
Annotations:       <none>
Selector:          app=nginx,lecture=k8
Type:              ClusterI
IP:                10.109.199.2
Port:              <unset>  80/TC
TargetPort:        8080/TCP
Endpoints:         172.17.0.24:8080
Session Affinity:  None
Events:            <none>
```

Service

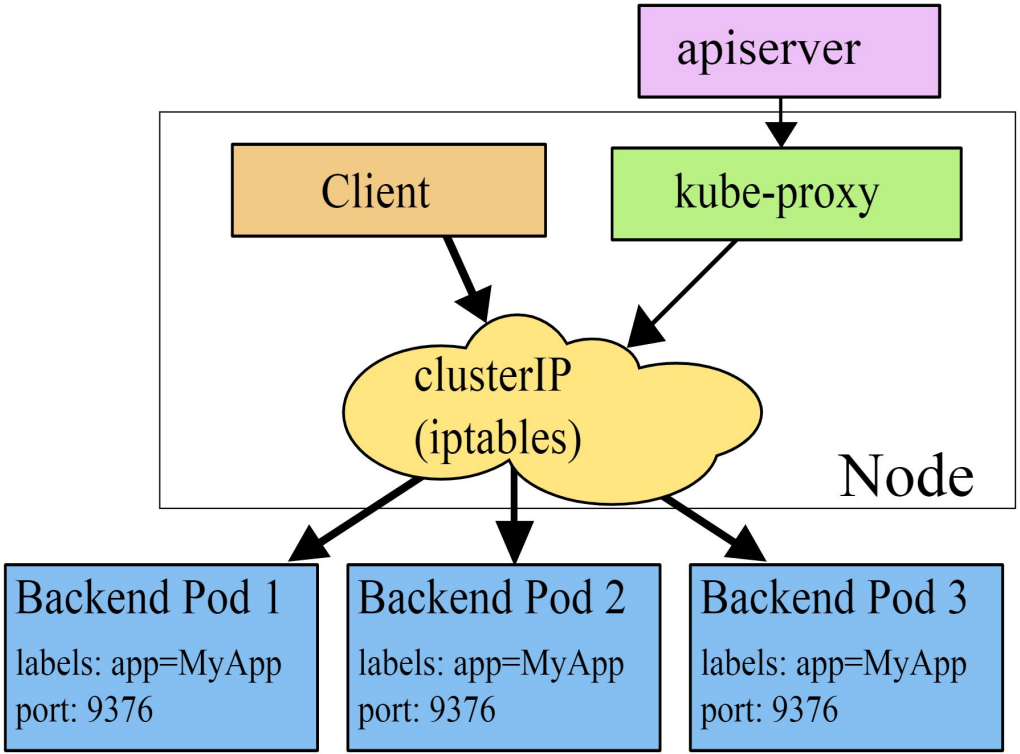
```
kubectl exec -it net-tools -- bash
bash-5.1# curl http://my-service:80/
Server address: 172.17.0.14:8080
Server name: nginx
Date: 09/Nov/2021:17:51:11 +0000
URI: /
Request ID: 7950531ce10e2747ab5fdb626f306b3b

bash-5.1# nslookup my-service
Server:         10.96.0.10
Address:        10.96.0.10#53

Name:   my-service.default.svc.cluster.local
Address: 10.109.199.28
```

Service

ITMO UNIVERSITY

```
kubectl exec -it net-tools -- bash
bash-5.1# curl http://my-service:80/
Server address: 172.17.0.14:8080
Server name: nginx
Date: 09/Nov/2021:17:51:11 +0000
URI: /
Request ID: 7950531ce10e2747ab5fdb626f306b3b

bash-5.1# nslookup my-service
Server:         10.96.0.10
Address:        10.96.0.10#53

Name:   my-service.default.svc.cluster.local
Address: 10.109.199.28

bash-5.1# ping -i 1
my-service
PING my-service.default.svc.cluster.local (10.109.199.28) 56(84) bytes of
data.
From ae1-461.RT.OV.SPB.RU.retn.net (87.245.250.78) icmp_seq=11 Destination Net
Unreachable
^C

--- my-service.default.svc.cluster.local ping statistics
---
21 packets transmitted, 0 received, +1 errors, 100% packet loss, time
20444ms
```

Service



https://kubernetes.io/docs/concepts/services-networking/service/

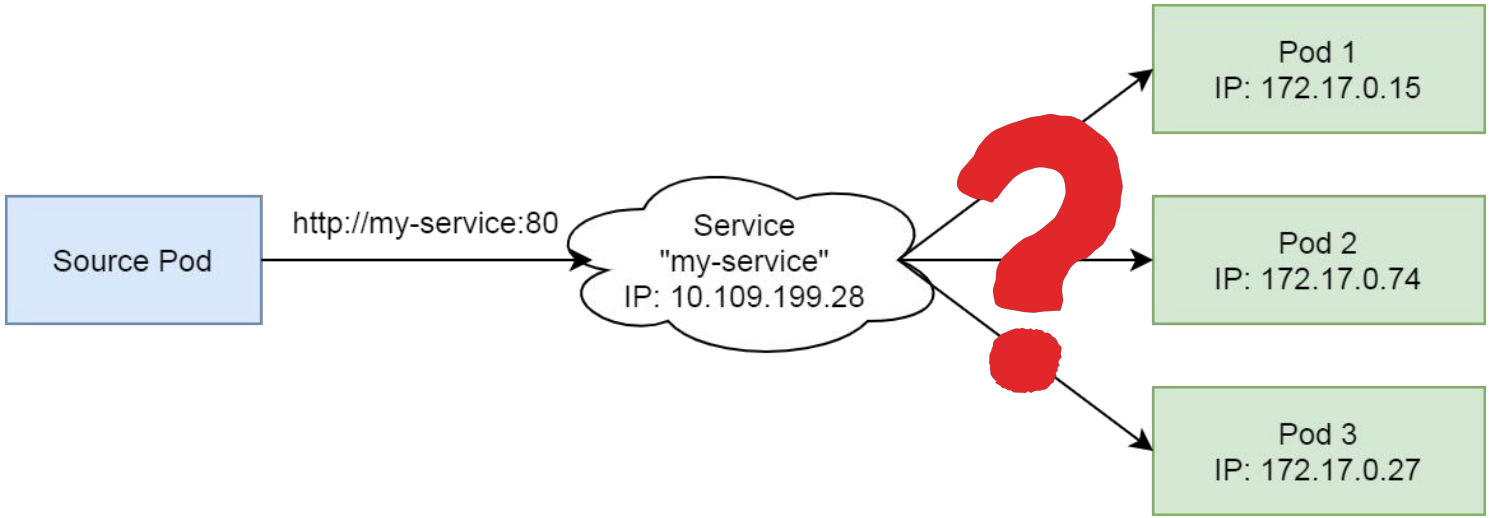
Service

ITMO UNIVERSITY

Service is NOT A PINGABLE object!

ITMOre than a UNIVERSITY

Service - Multiple Pods



Service - Multiple Pods

```
kubectl get pods -o wide
NAME        READY   STATUS    RESTARTS   AGE     IP             NODE       NOMINATED NODE   READINESS GAT
net-tools   1/1     Running   0          7h32m   172.17.0.23    host-1     <none>           <nor
nginx-1     1/1     Running   0          3m6s    172.17.0.15    host-1     <none>           <nor
nginx-2     1/1     Running   0          3m49s   172.17.0.24    host-1     <none>           <nor
nginx-3     1/1     Running   0          3m42s   172.17.0.25    host-1     <none>           <nor
```

Service - Multiple Pods

```
kubectl get pods -o wide
NAME        READY   STATUS    RESTARTS   AGE     IP             NODE       NOMINATED NODE   READINESS GA
net-tools   1/1     Running   0          7h32m   172.17.0.23    host-1     <none>           <nor
nginx-1     1/1     Running   0          3m6s    172.17.0.15    host-1     <none>           <nor
nginx-2     1/1     Running   0          3m49s   172.17.0.24    host-1     <none>           <nor
nginx-3     1/1     Running   0          3m42s   172.17.0.25    host-1     <none>           <nor
```

```
kubectl describe service my-service
Name:              my-servic
Namespace:         defaul
Labels:            lecture=k8
Annotations:       <none
Selector:          app=nginx,lecture=k8
Type:              ClusterI
IP:                10.109.199.2
Port:              <unset>  80/TC
TargetPort:        8080/TC
Endpoints:         172.17.0.15:8080,172.17.0.24:8080,172.17.0.25:808
Session Affinity:  Non
Events:            <none
```

Service - Multiple Pods

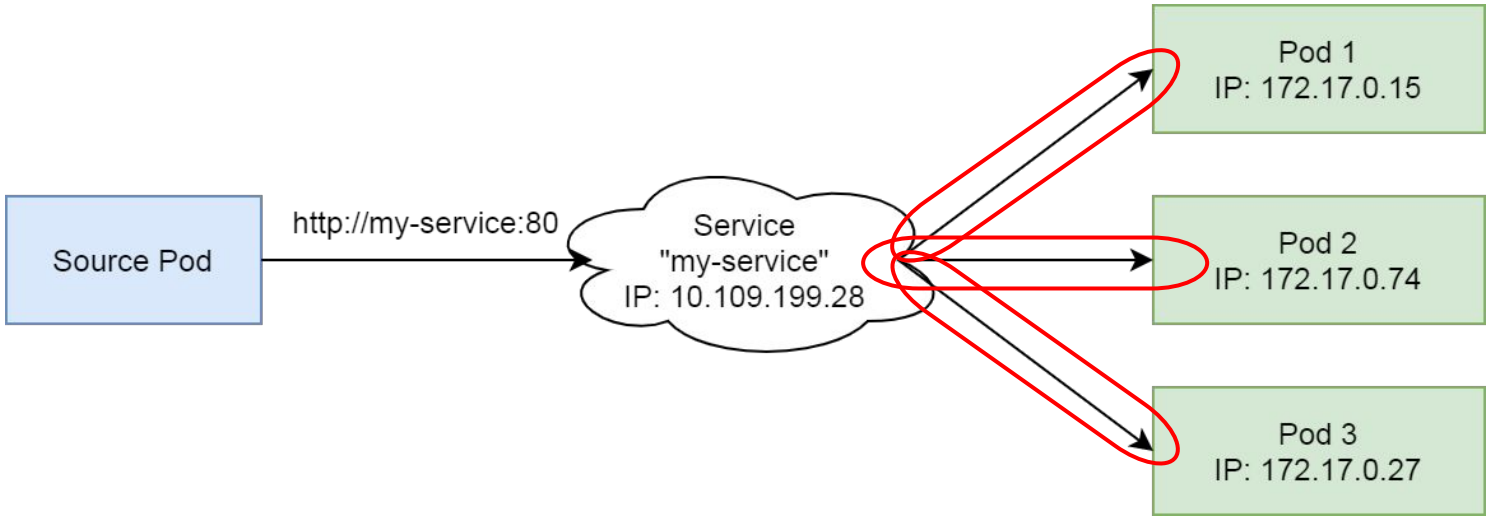
ITMO UNIVERSITY

```
kubectl exec -it net-tools -- bash
bash-5.1# for i in `seq 1 5`; do curl http://my-service:80; done
Server address: 172.17.0.25:8080
Server name: nginx-3
Date: 09/Nov/2021:20:00:24 +000
URI: /
Request ID: cc5dfd6deacc5f40ee69b66f775acbf
Server address: 172.17.0.25:8080
Server name: nginx-3
Date: 09/Nov/2021:20:00:24 +000
URI: /
Request ID: eafd91d41f2f77f8bc1d8928c82cf88
Server address: 172.17.0.24:8080
Server name: nginx-2
Date: 09/Nov/2021:20:00:24 +000
URI: /
Request ID: ff516cff3b5dda22181f16682c77673
Server address: 172.17.0.15:8080
Server name: nginx-1
Date: 09/Nov/2021:20:00:24 +000
URI: /
Request ID: 978821f3932d4e0526fa4f91cccad52
Server address: 172.17.0.15:8080
Server name: nginx-1
Date: 09/Nov/2021:20:00:24 +000
URI: /
Request ID: 204eb6f66de7586ce1f039d8f4febd0
```

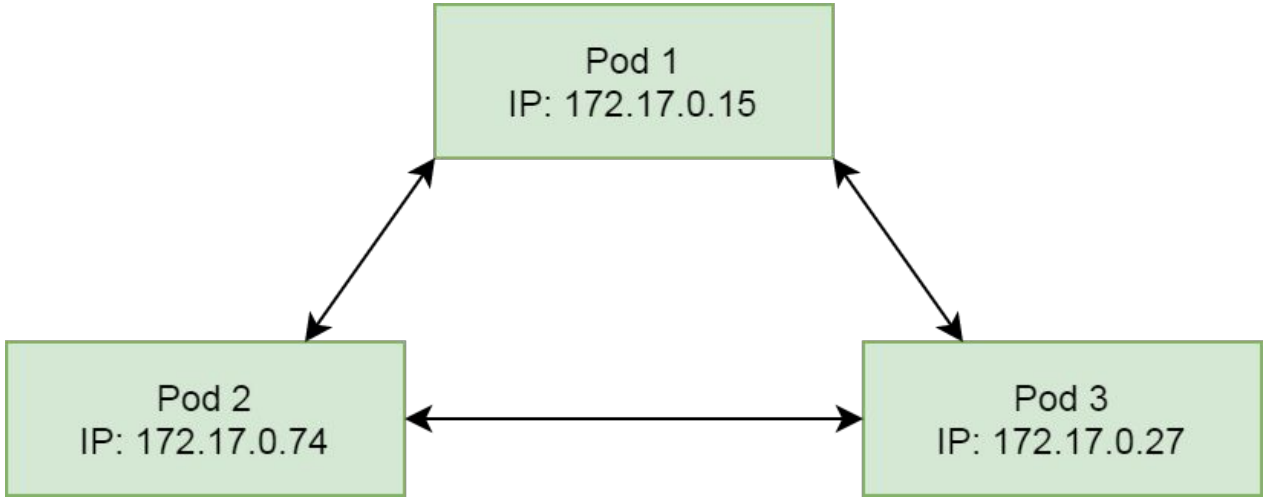
```
bash-5.1# nslookup my-service
Server:         10.96.0.10
Address:        10.96.0.10#53

Name:
my-service.default.svc.cluster.local
Address: 10.109.199.28
```

UNIVERSITY

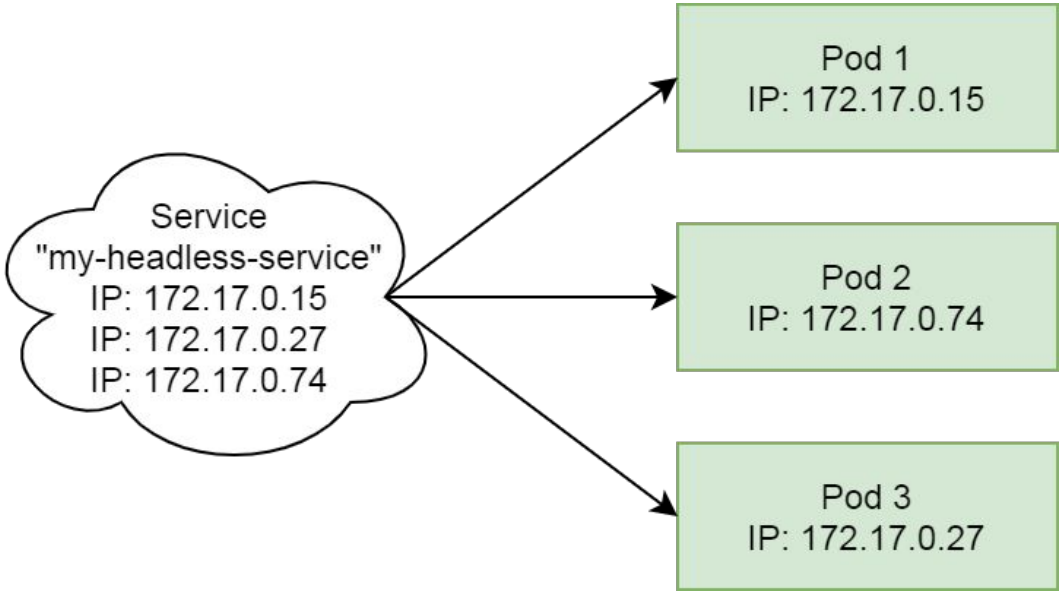
Service - Multiple Pods



Headless Service



Headless Service



Headless Service

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Service
metadata:
  name: headless-service
  labels:
    lecture: k8s
spec:
  clusterIP: None
  ports:
    - port: 80
      targetPort: 8080
  selector:
    app: nginx
    lecture: k8s
```

ITMO more than a UNIVERSITY

Headless Service

ITMO UNIVERSITY

```yaml
apiVersion: v1
kind: Service
metadata:
  name: headless-service
  labels:
    lecture: k8s
spec:
  clusterIP: None
  ports:
    - port: 80
      targetPort: 8080
  selector:
    app: nginx
    lecture: k8s
```

ITMO more than a UNIVERSITY

Headless Service

```
kubectl apply -f headless-service.yaml
service/headless-service create
```

```
kubectl get services
NAME              TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    A
headless-service  ClusterIP   None            <none>        80/TCP     6
kubernetes        ClusterIP   10.96.0.1       <none>        443/TCP    7d
my-service        ClusterIP   10.109.199.28   <none>        80/TCP
```

```
kubectl describe service headless-service
Name:              headless-servic
Namespace:         defaul
Labels:            lecture=k8
Annotations:       <none>
Selector:          app=nginx,lecture=k8
Type:              ClusterI
IP:                None
Port:              <unset>  80/TC
TargetPort:        8080/TC
Endpoints:         172.17.0.15:8080,172.17.0.24:8080,172.17.0.25:80
Session Affinity:  None
Events:            <none>
```

Headless Service

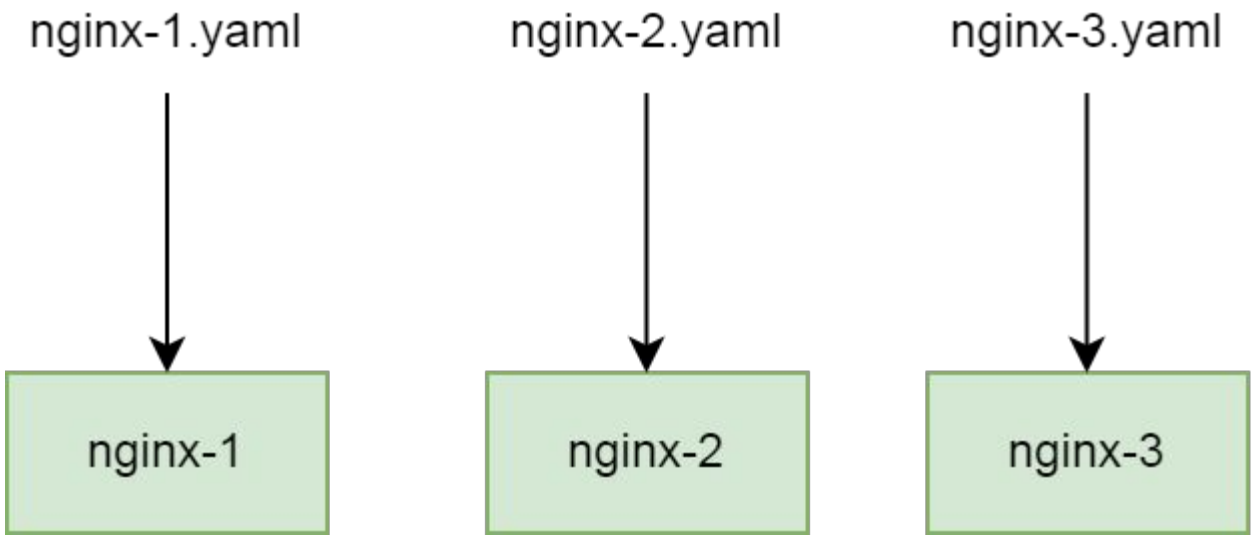
```
kubectl exec -it net-tools -- bash
bash-5.1# nslookup headless-servic
Server:         10.96.0.1
Address:        10.96.0.10#5

Name:   headless-service.default.svc.cluster.loca
Address: 172.17.0.2
Name:   headless-service.default.svc.cluster.loca
Address: 172.17.0.2
Name:   headless-service.default.svc.cluster.loca
Address: 172.17.0.1

bash-5.1# nslookup my-service
Server:         10.96.0.1
Address:        10.96.0.10#5

Name:   my-service.default.svc.cluster.loca
Address: 10.109.199.2
```

Headless Service

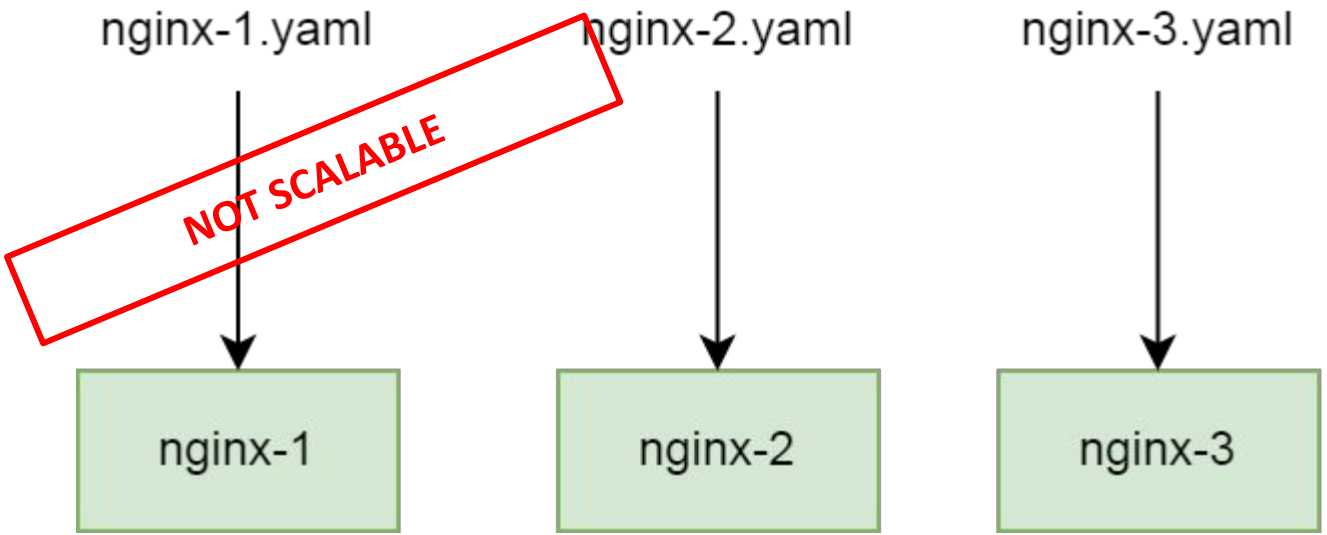
```
kubectl exec -it net-tools -- bash
bash-5.1# nslookup headless-servic
Server:         10.96.0.1
Address:        10.96.0.10#5

Name:   headless-service.default.svc.cluster.loca
Address: 172.17.0.2
Name:   headless-service.default.svc.cluster.loca
Address: 172.17.0.24
Name:   headless-service.default.svc.cluster.loca
Address: 172.17.0.1

bash-5.1# nslookup my-servic
Server:         10.96.0.1
Address:        10.96.0.10#5

Name:   my-service.default.svc.cluster.loca
Address: 10.109.199.2
```

```
kubectl exec -it net-tools -- bash
bash-5.1# curl http://my-service:80
Server address: 172.17.0.24:808
Server name: nginx-
Date: 09/Nov/2021:21:53:47 +000
URI: /
Request ID: 2b92b8b4dce82052722f82ad8b2df73

bash-5.1# curl http://my-service:8080/ --max-time 1
curl: (28) Connection timed out after 10001 millisecond

bash-5.1# curl http://headless-service:80
curl: (7) Failed to connect to headless-service port 8
after 1 ms: Connection refuse

bash-5.1# curl http://headless-service:8080
Server address: 172.17.0.15:808
Server name: nginx-
Date: 09/Nov/2021:22:00:44 +000
URI: /
Request ID: 0a9aaee8dddb0d9885e54fdd51fcb06
```

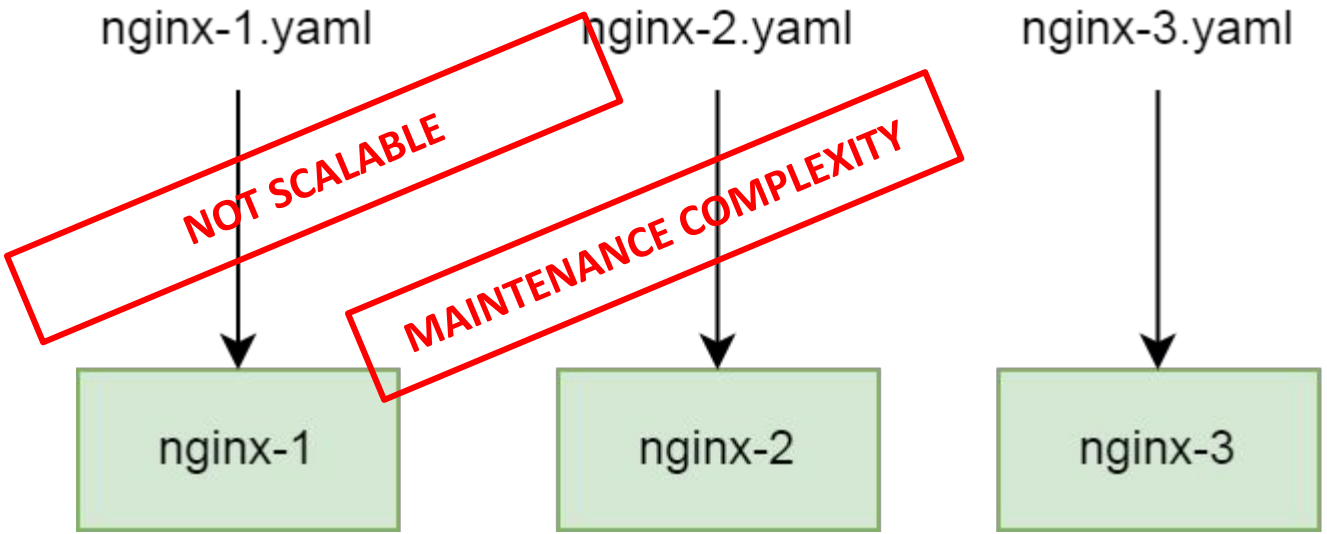
Pod replicas



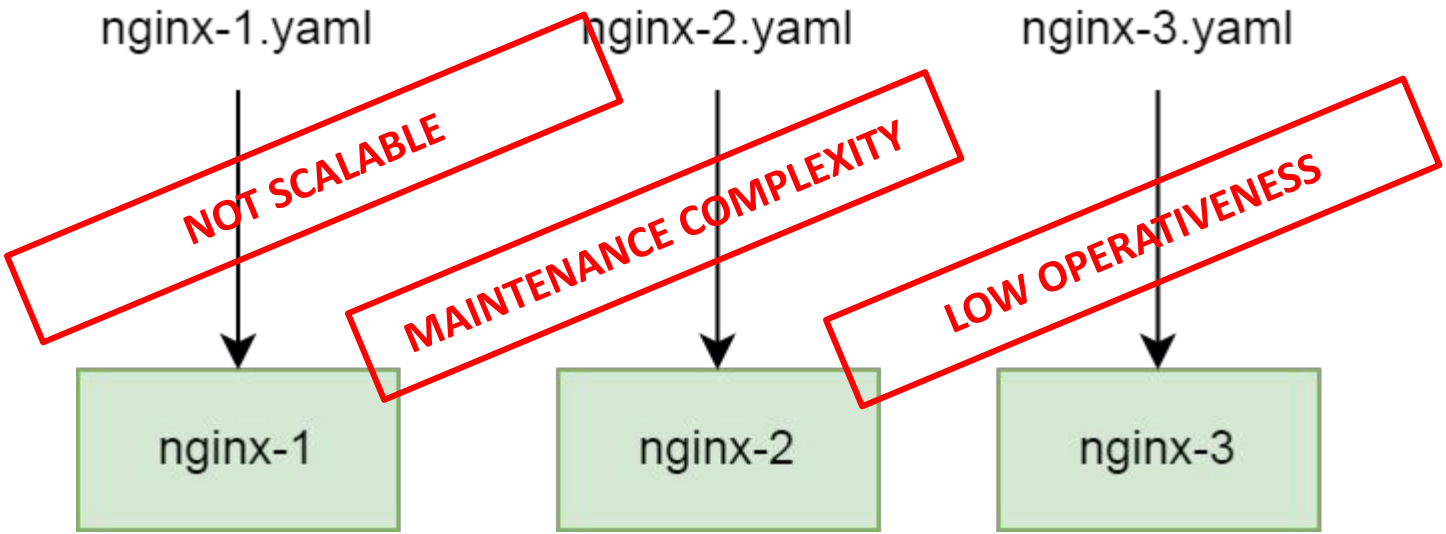
Pod replicas



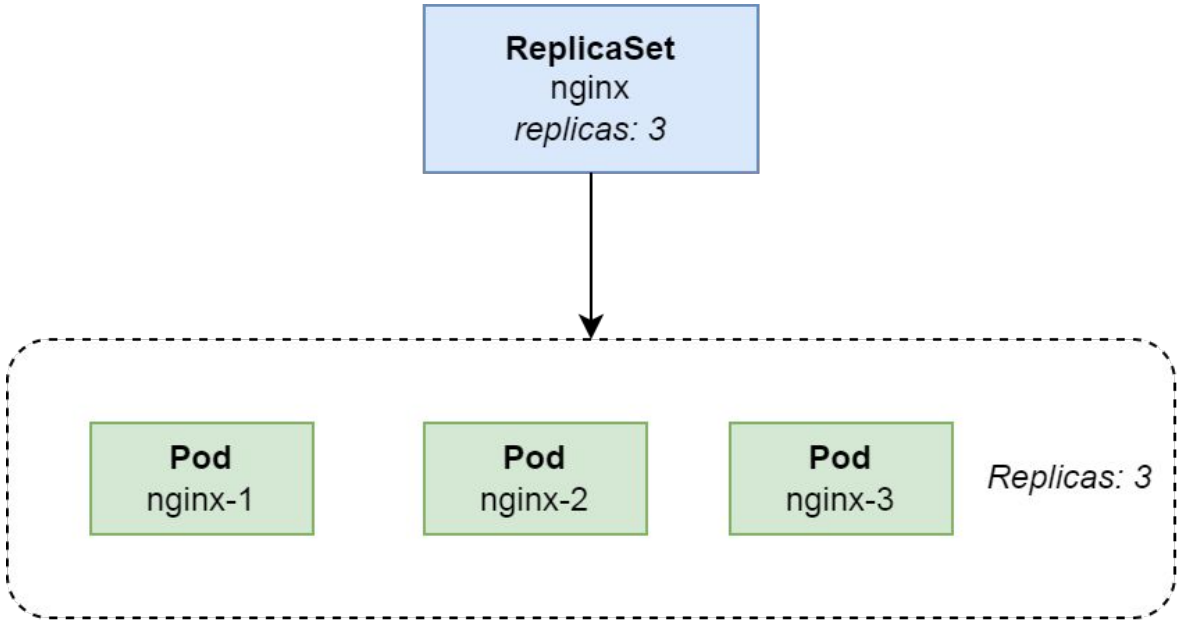
Pod replicas



Pod replicas



ReplicaSet


ReplicaSet
nginx
replicas: 3
Pod
nginx-1
Pod
nginx-2
Pod
nginx-3
Replicas: 3

ReplicaSet

ITMO UNIVERSITY

```yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-replicaset
      lecture: k8s
  template:
    metadata:
      labels:
        app: nginx-replicaset
        lecture: k8s
    spec:
      containers:
      - name: nginx
        image: nginx:1.17
```

ITMO more than a UNIVERSITY

ReplicaSet

ITMO UNIVERSITY

```yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-replicaset
      lecture: k8s
  template:
    metadata:
      labels:
        app: nginx-replicaset
        lecture: k8s
    spec:
      containers:
      - name: nginx
        image: nginx:1.17
```

ITMO more than a UNIVERSITY

ReplicaSet

```yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-replicaset
      lecture: k8s
  template:
    metadata:
      labels:
        app: nginx-replicaset
        lecture: k8s
    spec:
      containers:
      - name: nginx
        image: nginx:1.17
```

ReplicaSet

ITMO UNIVERSITY

```yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-replicaset
      lecture: k8s
  template:
    metadata:
      labels:
        app: nginx-replicaset
        lecture: k8s
    spec:
      containers:
      - name: nginx
        image: nginx:1.17
```

ITMO more than a UNIVERSITY

ReplicaSet

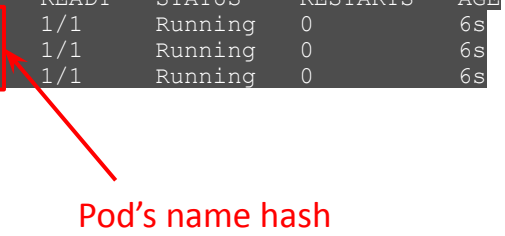
ITMO UNIVERSITY

```yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-replicaset
      lecture: k8s
  template:
    metadata:
      labels:
        app: nginx-replicaset
        lecture: k8s
    spec:
      containers:
      - name: nginx
        image: nginx:1.17
```

ITMO more than a UNIVERSITY

ReplicaSet

```
kubectl apply -f nginx-replicaset.yaml
replicaset.apps/nginx created

kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-2xs5f   1/1     Running   0          6s
nginx-flv4j   1/1     Running   0          6s
nginx-rt29l   1/1     Running   0          6s
```



ReplicaSet

```
kubectl apply -f nginx-replicaset.yaml
replicaset.apps/nginx created

kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-2xs5f   1/1     Running   0          6s
nginx-flv4j   1/1     Running   0          6s
nginx-rt29l   1/1     Running   0          6s

kubectl describe pod nginx-2xs5f
Name:         nginx-2xs5f
Namespace:    default
Priority:     0
Node:         host-1/<node-IP>
Start Time:   Sun, 28 Nov 2021 20:13:52 +0300
Labels:       app=nginx-replicaset
              lecture=k8s
Annotations:  <none>
Status:       Running
IP:           172.17.0.25
IPs:
  IP:           172.17.0.25
Controlled By:  ReplicaSet/nginx
```

ReplicaSet

```
kubectl apply -f nginx-replicaset.yaml
replicaset.apps/nginx created

kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-2xs5f   1/1     Running   0          6s
nginx-flv4j   1/1     Running   0          6s
nginx-rt29l   1/1     Running   0          6s

kubectl describe pod nginx-2xs5f
Name:         nginx-2xs5f
Namespace:    default
Priority:     0
Node:         host-1/<node-IP>
Start Time:   Sun, 28 Nov 2021 20:13:52 +0300
Labels:       app=nginx-replicaset
              lecture=k8s
Annotations:  <none>
Status:       Running
IP:           172.17.0.25
IPs:
  IP:           172.17.0.25
Controlled By:  ReplicaSet/nginx
```

```
kubectl get rs
NAME    DESIRED   CURRENT   READY   AGE
nginx   3         3         3       12s

kubectl describe rs nginx
Name:
nginx
Namespace:
default
Selector:
app=nginx-replicaset,lecture=k8s
Labels:
lecture=k8s
Annotations:
<none>
Replicas:     3 current / 3
desired
Pods Status:  3 Running / 0 Waiting / 0 Succeeded / 0
Failed
Pod
Template:
  Labels:
app=nginx-replicaset

lecture=k8s

Containers:

nginx:
    Image:
nginx:1.17
    Port:
<none>
    Host Port:
<none>
    Environment:
<none>
    Mounts:
<none>
```

ReplicaSet

```
kubectl apply -f nginx-replicaset.yaml
replicaset.apps/nginx configured
```

```
kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-2xs5f   1/1     Running   0          28m
nginx-b6wn5   1/1     Running   0          4s
nginx-flv4j   1/1     Running   0          28m
nginx-pf992   1/1     Running   0          4s
nginx-rt29l   1/1     Running   0          28m
nginx-xbl65   1/1     Running   0          4s
```

```
kubectl get rs
NAME    DESIRED   CURRENT   READY   AGE
nginx   6         6         6       28m
```

```
kubectl describe rs nginx
Name:
nginx
Namespace:
default
Selector:
app=nginx-replicaset,lecture=k8s
Labels:
lecture=k8s
Annotations:
<none>
Replicas:       6 current /  6
desired
Pods Status:    6 Running / 0 Waiting / 0 Succeeded / 0
Failed
Pod
Template:
  Labels:
app=nginx-replicaset

lecture=k8s

Containers:

nginx:
    Image:
nginx:1.17
    Port:
<none>
    Host Port:
<none>
    Environment:
<none>
    Mounts:
<none>
```

ReplicaSet

```
kubectl delete pod nginx-2xs5f
pod "nginx-2xs5f" deleted

kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-b6wn5   1/1     Running   0          9m2s
nginx-flv4j   1/1     Running   0          37m
nginx-pf992   1/1     Running   0          9m2s
nginx-rk869   1/1     Running   0          7s
nginx-rt291   1/1     Running   0          37m
nginx-xbl65   1/1     Running   0          9m2s
```

```
kubectl get rs
NAME    DESIRED   CURRENT   READY   AGE
nginx   6         6         6       28m

kubectl describe rs nginx
Name:
nginx
Namespace:
default
Selector:
app=nginx-replicaset,lecture=k8s
Labels:
lecture=k8s
Annotations:
<none>
Replicas:     6 current / 6
desired
Pods Status:  6 Running / 0 Waiting / 0 Succeeded / 0
Failed
Pod
Template:
  Labels:
app=nginx-replicaset

lecture=k8s

Containers:

nginx:
    Image:
nginx:1.17
    Port:
<none>
    Host Port:
<none>
    Environment:
<none>
    Mounts:
<none>
```

ReplicaSet

```
kubectl apply -f nginx-replicaset.yaml
replicaset.apps/nginx configured
```

```
kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-b6wn5   1/1     Running   0          80m
nginx-flv4j   1/1     Running   0          109m
nginx-pf992   1/1     Running   0          80m
nginx-rk869   1/1     Running   0          71m
nginx-rt29l   1/1     Running   0          109m
nginx-xbl65   1/1     Running   0          80m
```

```
kubectl describe pod nginx-2xs5f
Name:         nginx-b6wn5
Namespace:    default
...
Containers:
  nginx:
    Container ID:   docker://<container-ID>
    Image:          nginx:1.17
```

```
kubectl get rs
NAME    DESIRED   CURRENT   READY   AGE
nginx   6         6         6       28m
```

```
kubectl describe rs nginx
Name:         nginx
Namespace:    default
Selector:     app=nginx-replicaset,lecture=k8s
Labels:       lecture=k8s
Annotations:  <none>
Replicas:     6 current / 6 desired
Pods Status:  6 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=nginx-replicaset
           lecture=k8s
  Containers:
   nginx:
    Image:        nginx:1.18
    Port:         <none>
    Host Port:    <none>
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Events:           <none>
```

ReplicaSet

```
kubectl delete pod nginx-b6wn5
pod "nginx-b6wn5" deleted
```

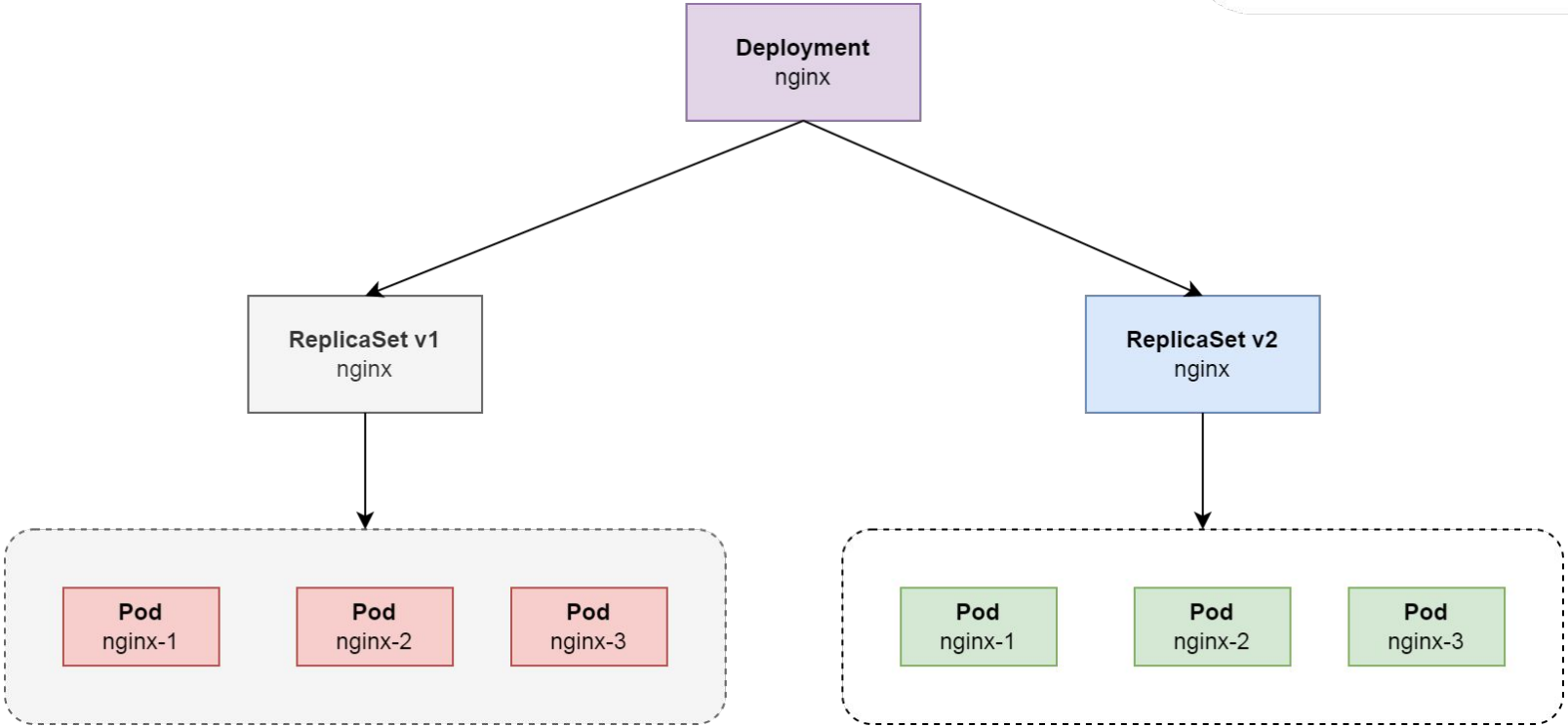
```
kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-flv4j   1/1     Running   0          121m
nginx-pf992   1/1     Running   0          92m
nginx-rk869   1/1     Running   0          83m
nginx-rt29l   1/1     Running   0          121m
nginx-wgb8b   1/1     Running   0          57s
nginx-xbl65   1/1     Running   0          92m
```

```
kubectl describe pod nginx-wgb8b
Name:         nginx-wgb8b
Namespace:    default
...
Containers:
  nginx:
    Container ID:   docker://<container-ID>
    Image:          nginx:1.18
```

```
kubectl get rs
NAME    DESIRED   CURRENT   READY   AGE
nginx   6         6         6       28m
```

```
kubectl describe rs nginx
Name:
nginx
Namespace:
default
Selector:
app=nginx-replicaset,lecture=k8s
Labels:
lecture=k8s
Annotations:
<none>
Replicas:     6 current / 6
desired
Pods Status:  6 Running / 0 Waiting / 0 Succeeded / 0
Failed
Pod
Template:
  Labels:
app=nginx-replicaset

lecture=k8s

Containers:

nginx:
    Image:
nginx:1.18
    Port:
<none>
    Host Port:
<none>
    Environment:
<none>
    Mounts:
<none>
```

Deployment


Deployment
nginx
ReplicaSet v1
nginx
ReplicaSet v2
nginx
Pod
nginx-1
Pod
nginx-2
Pod
nginx-3
Pod
nginx-1
Pod
nginx-2
Pod
nginx-3

Deployment

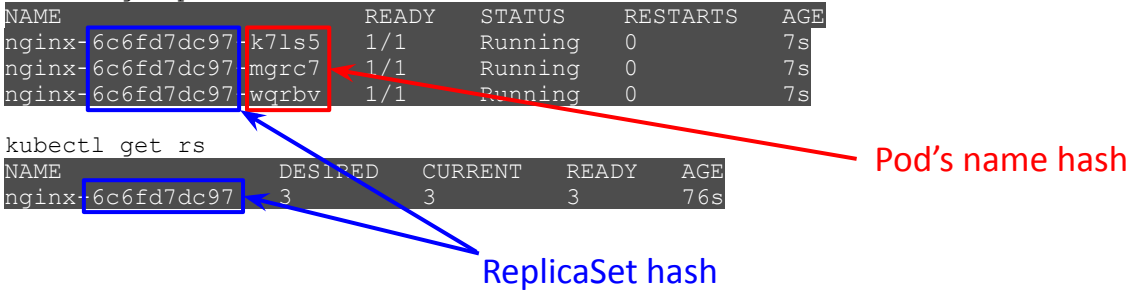
ITMO UNIVERSITY

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx-deployment
      lecture: k8s
  template:
    metadata:
      labels:
        app: nginx-deployment
        lecture: k8s
    spec:
      containers:
      - name: nginx
        image: nginx:1.17
```

ITMO more than a UNIVERSITY

Deployment

```
kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx created

kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
nginx-6c6fd7dc97-k7ls5   1/1     Running   0          7s
nginx-6c6fd7dc97-mgrc7   1/1     Running   0          7s
nginx-6c6fd7dc97-wqrbv   1/1     Running   0          7s

kubectl get rs
NAME               DESIRED   CURRENT   READY   AGE
nginx-6c6fd7dc97   3         3         3       76s
```



Deployment

```
kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx created

kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
nginx-6c6fd7dc97-k7ls5   1/1     Running   0          7s
nginx-6c6fd7dc97-mgrc7   1/1     Running   0          7s
nginx-6c6fd7dc97-wqrbv   1/1     Running   0          7s

kubectl get rs
NAME               DESIRED   CURRENT   READY   AGE
nginx-6c6fd7dc97   3         3         3       76s

kubectl describe pod nginx-6c6fd7dc97-wqrbv
Name:           nginx-6c6fd7dc97-wqrbv
Namespace:      default
Priority:       0
Node:           host-1/<host-IP>
Start Time:     Mon, 29 Nov 2021 00:03:28 +0300
Labels:         app=nginx-deployment
                lecture=k8s
                pod-template-hash=6c6fd7dc97
Annotations:    <none>
Status:         Running
IP:             172.17.0.24
IPs:
  IP:           172.17.0.24
Controlled By:  ReplicaSet/nginx-6c6fd7dc97
```

Deployment

```
kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx created

kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
nginx-6c6fd7dc97-k7ls5   1/1     Running   0          7s
nginx-6c6fd7dc97-mgrc7   1/1     Running   0          7s
nginx-6c6fd7dc97-wqrbv   1/1     Running   0          7s

kubectl get rs
NAME               DESIRED   CURRENT   READY   AGE
nginx-6c6fd7dc97   3         3         3       76s

kubectl describe pod nginx-6c6fd7dc97-wqrbv
Name:           nginx-6c6fd7dc97-wqrbv
Namespace:      default
Priority:       0
Node:           host-1/<host-IP>
Start Time:     Mon, 29 Nov 2021 00:03:28 +0300
Labels:         app=nginx-deployment
                lecture=k8s
                pod-template-hash=6c6fd7dc97
Annotations:    <none>
Status:         Running
IP:             172.17.0.24
IPs:
  IP:           172.17.0.24
Controlled By:  ReplicaSet/nginx-6c6fd7dc97
```

```
kubectl describe rs nginx-6c6fd7dc97
Name:           nginx-6c6fd7dc97
Namespace:      default
Selector:
app=nginx-deployment,lecture=k8s, pod-template-hash=6c6fd7dc97
Labels:         app=nginx-deployment
                lecture=k8s
                pod-template-hash=6c6fd7dc97
Annotations:    deployment.kubernetes.io/desired-replicas: 3
                deployment.kubernetes.io/max-replicas: 4
                deployment.kubernetes.io/revision: 1
Controlled By:  Deployment/nginx
Replicas:       3 current / 3 desired
Pods Status:    3 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=nginx-deployment
           lecture=k8s
           pod-template-hash=6c6fd7dc97
```

Deployment

```
kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx configured

kubectl get pods
NAME                     READY   STATUS        RESTARTS   AGE
nginx-6c6fd7dc97-k7ls5   0/1     Terminating   0          117m
nginx-6c6fd7dc97-mgrc7   0/1     Terminating   0          117m
nginx-7db44bd9f6-cjjck   1/1     Running       0          9s
nginx-7db44bd9f6-kdd2j   1/1     Running       0          12s
nginx-7db44bd9f6-pwptb   1/1     Running       0          6s

kubectl get rs
NAME               DESIRED   CURRENT   READY   AGE
nginx-6c6fd7dc97   0         0         0       119m
nginx-7db44bd9f6   3         3         3       107s
```

Deployment

```
kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx configured

kubectl get pods
NAME                     READY   STATUS        RESTARTS   AGE
nginx-6c6fd7dc97-k7ls5   0/1     Terminating   0          117m
nginx-6c6fd7dc97-mgrc7   0/1     Terminating   0          117m
nginx-7db44bd9f6-cjjck   1/1     Running       0          9s
nginx-7db44bd9f6-kdd2j   1/1     Running       0          12s
nginx-7db44bd9f6-pwptb   1/1     Running       0          6s

kubectl get rs
NAME               DESIRED   CURRENT   READY   AGE
nginx-6c6fd7dc97   0         0         0       119m
nginx-7db44bd9f6   3         3         3       107s
```

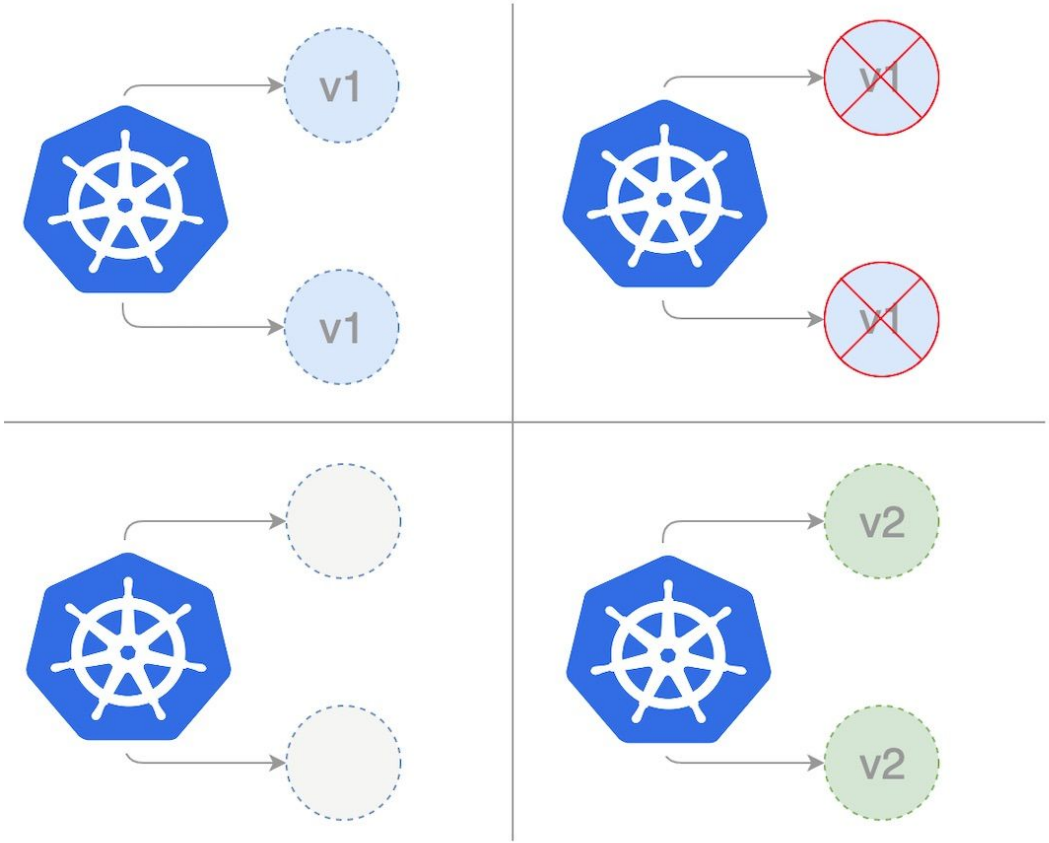
Out of the box strategies:

- Rolling Update (default)
- Recreate

Advanced strategies:

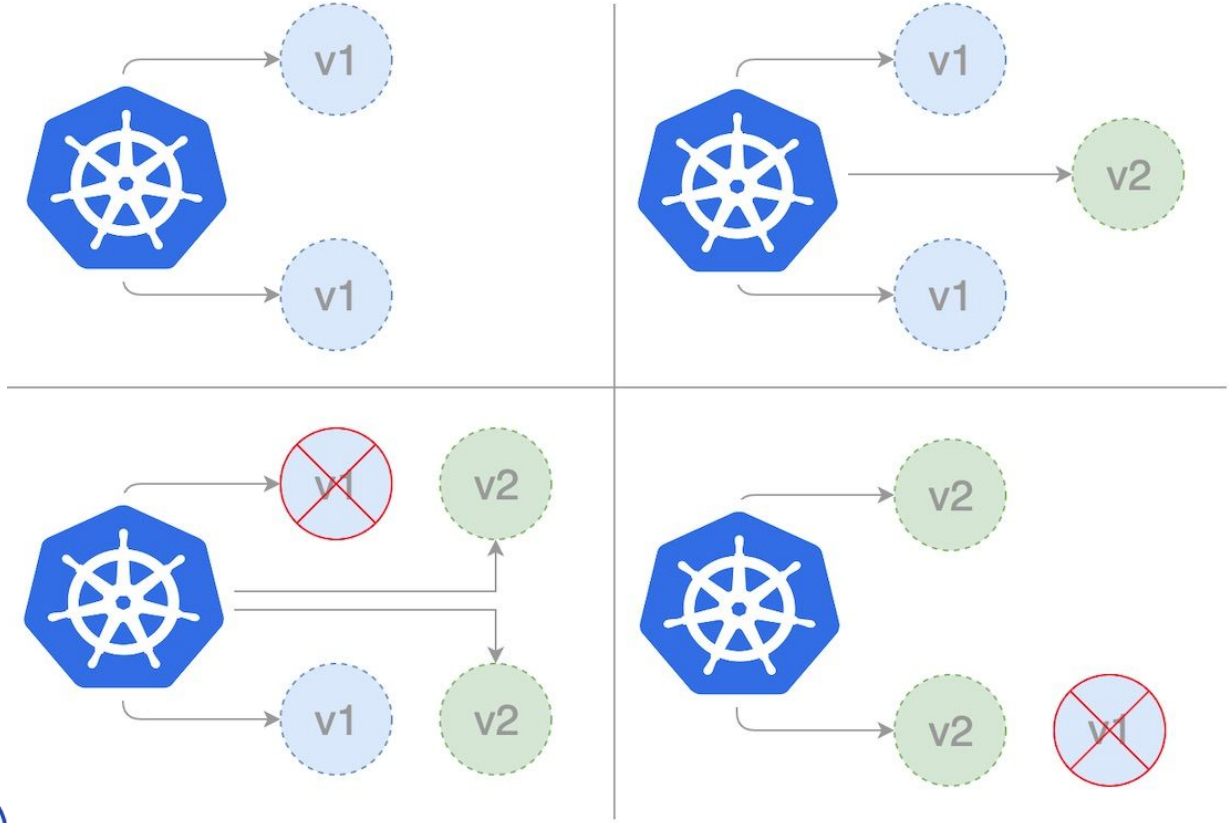
- Canary
- A/B
- Blue-Green
- ...other

Deployment: Recreate



https://auth0.com/blog/deployment-strategies-in-kubernetes/

Deployment: Rolling Update



https://auth0.com/blog/deployment-strategies-in-kubernetes/

Deployment

ITMO UNIVERSITY

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  replicas: 10
  strategy:
    type: RollingUpdate  # Recreate
    rollingUpdate:
      maxSurge: 3  # 0.1 i.e. 10%
      maxUnavailable: 2  # 0.25 i.e. 25%
  selector:
    matchLabels:
      app: nginx-deployment
      lecture: k8s
  template:
    ...
```

ITMO more than a UNIVERSITY

Deployment

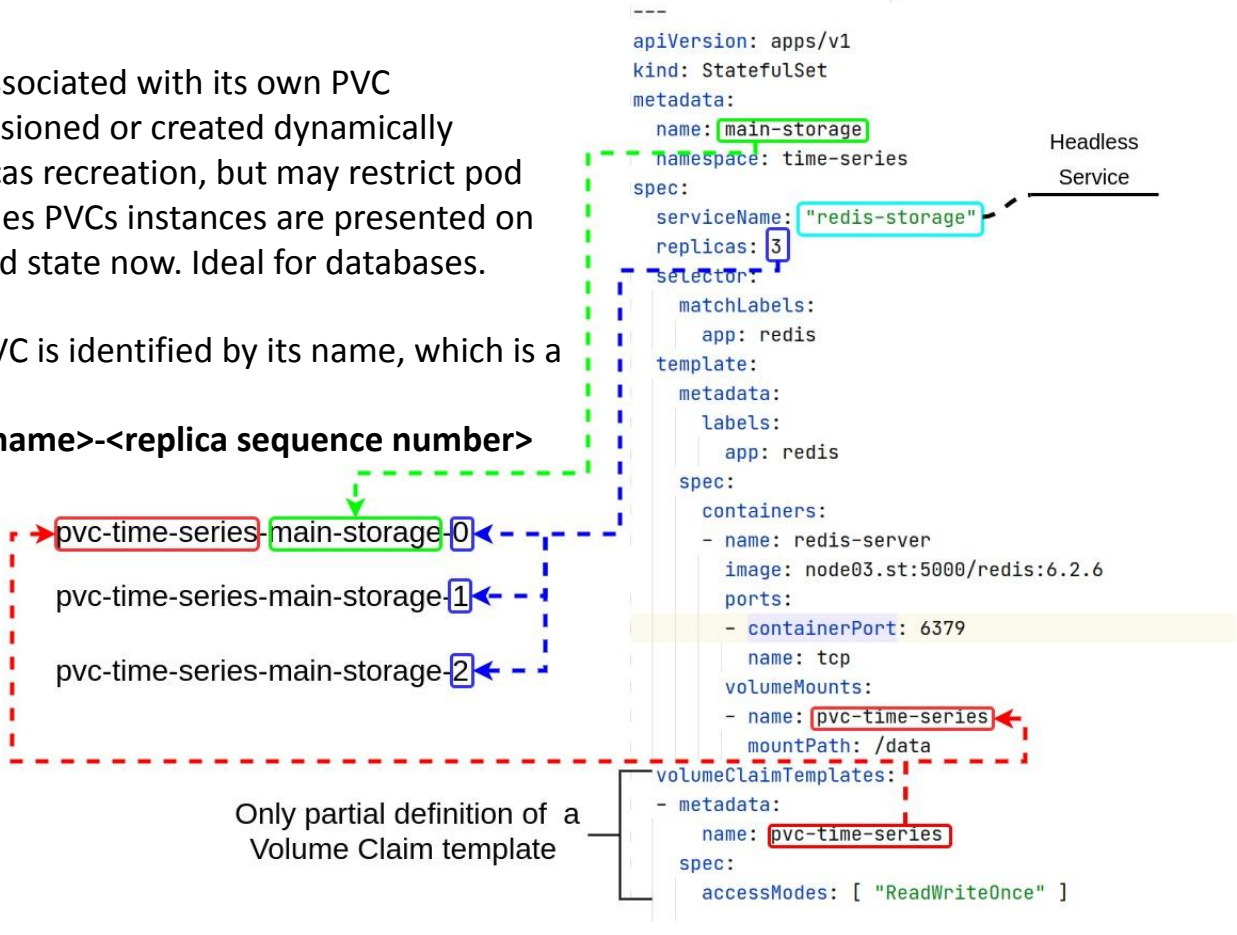
```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  replicas: 10
  strategy:
    type: RollingUpdate
    rollingUpdate:
      maxSurge: 3
      maxUnavailable: 2
  selector:
    matchLabels:
      app: nginx-deployment
      lecture: k8s
  template:
    ...
```

```
Iteration 1:
     NEW: 0 -> 3 (+ maxSurge)
     OLD: 10 -> 8 (- maxUnavailable)
```

StatefulSet

- Replica-based too...
- ...but each replica is associated with its own PVC
- PVCs may be pre-provisioned or created dynamically
- They will survive replicas recreation, but may restrict pod scheduling only to nodes PVCs instances are presented on
- At the end: we can hold state now. Ideal for databases.

For each replica, a required PVC is identified by its name, which is a combination:
<name of VCT>-<StatefulSet name>-<replica sequence number>



StatefulSet: headless service

ITMO UNIVERSITY

- Headless service creates DNS records for all replicas in a StatefulSet
- One can reach an individual POD of replicas using a particular DNS name

```
---
apiVersion: v1
kind: Service
metadata:
  namespace: time-series
  name: redis-storage
  labels:
    app: redis
spec:
  ports:
  - port: 6379
    name: tcp
  clusterIP: None
  selector:
    app: redis
```

main-storage-0.redis-storage.time-series.svc.cluster.local

main-storage-1.redis-storage.time-series.svc.cluster.local

main-storage-2.redis-storage.time-series.svc.cluster.local

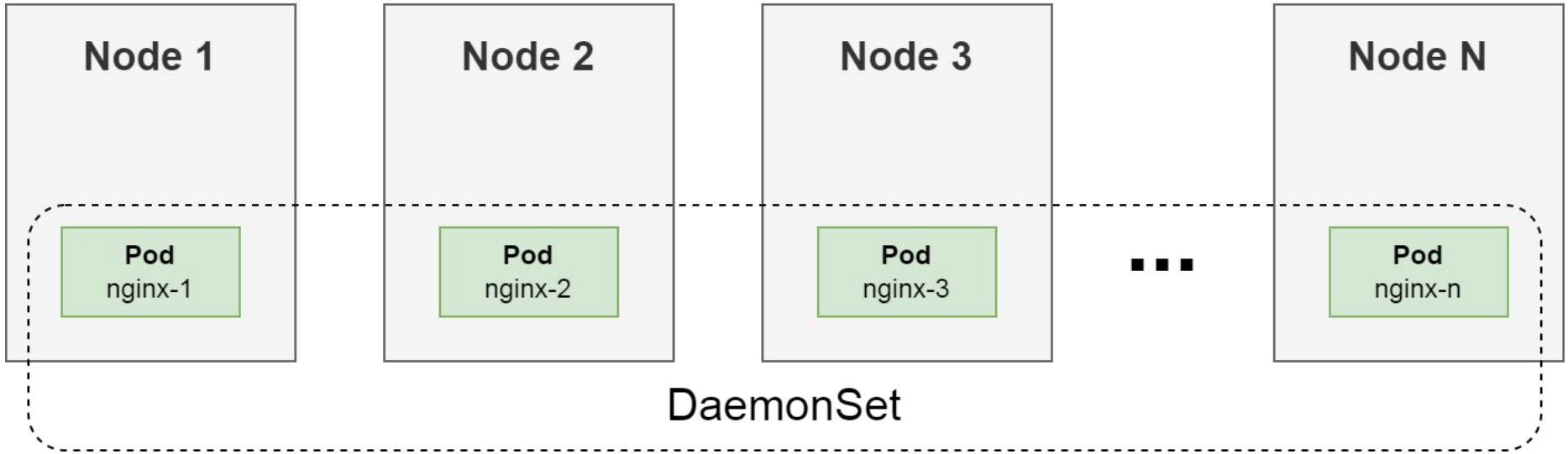
ITMO more than a UNIVERSITY

StatefulSet with headless service

ITMO UNIVERSITY

```
(default) [nikolay@localhost bigdatacourse]$  kubectl -n time-series run -i -t nwtool --image node03.st:5000/praqma/network
-multitool --restart=Never --force=true --rm=true --command /bin/bash
If you don't see a command prompt, try pressing enter.
bash-5.1# nslookup redis-storage
Server:         10.129.0.10
Address:        10.129.0.10#53

Name:   redis-storage.time-series.svc.cluster.local
Address: 10.128.194.181
Name:   redis-storage.time-series.svc.cluster.local
Address: 10.128.232.58
Name:   redis-storage.time-series.svc.cluster.local
Address: 10.128.251.22

bash-5.1# nslookup main-storage-0.redis-storage
Server:         10.129.0.10
Address:        10.129.0.10#53

Name:   main-storage-0.redis-storage.time-series.svc.cluster.local
Address: 10.128.232.58

bash-5.1# nslookup main-storage-1.redis-storage
Server:         10.129.0.10
Address:        10.129.0.10#53

Name:   main-storage-1.redis-storage.time-series.svc.cluster.local
Address: 10.128.194.181

bash-5.1# nslookup main-storage-2.redis-storage
Server:         10.129.0.10
Address:        10.129.0.10#53

Name:   main-storage-2.redis-storage.time-series.svc.cluster.local
Address: 10.128.251.22
```

DaemonSet


Node 1
Node 2
Node 3
Node N
Pod
nginx-1
Pod
nginx-2
Pod
nginx-3
Pod
nginx-n
DaemonSet

DaemonSet

ITMO UNIVERSITY

```yaml
apiVersion: apps/v1
kind: DaemonSet
metadata:
  name: nginx
  labels:
    lecture: k8s
spec:
  selector:
    matchLabels:
      app: nginx-daemonset
      lecture: k8s
  template:
    metadata:
      labels:
        app: nginx-daemonset
        lecture: k8s
    spec:
      containers:
      - name: nginx
        image: nginx:1.17
```

ITMO more than a UNIVERSITY

DaemonSet

ITMO UNIVERSITY

```
kubectl apply -f nginx-daemonset.yaml
daemonset.apps/nginx created

kubectl get pods -o=wide
NAME          READY   STATUS    RESTARTS   AGE   IP              NODE        NOMINATED NODE   READINESS GATES
nginx-4xrmp   1/1     Running   0          8s    10.128.75.151   node60.st   <none>           <none>
nginx-52bd8   1/1     Running   0          8s    10.128.120.135  node54.st   <none>           <none>
nginx-5klbb   1/1     Running   0          7s    10.128.251.136  node13.st   <none>           <none>
nginx-6ft8w   1/1     Running   0          8s    10.128.193.98   node47.st   <none>           <none>
nginx-6mb2m   1/1     Running   0          8s    10.128.23.198   node09.st   <none>           <none>
nginx-6t6r9   1/1     Running   0          8s    10.128.37.115   node28.st   <none>           <none>
nginx-7lcml   1/1     Running   0          8s    10.128.1.39     node25.st   <none>           <none>
nginx-7z5xk   1/1     Running   0          7s    10.128.229.199  node53.st   <none>           <none>
nginx-8dsxj   1/1     Running   0          7s    10.128.232.32   node15.st   <none>           <none>
nginx-8fdnc   1/1     Running   0          8s    10.128.17.143   node05.st   <none>           <none>
nginx-8t44c   1/1     Running   0          8s    10.128.17.18    node41.st   <none>           <none>
nginx-94mrz   1/1     Running   0          8s    10.128.201.85   node24.st   <none>           <none>
nginx-9hd5j   1/1     Running   0          8s    10.128.251.52   node16.st   <none>           <none>
nginx-9tjkz   1/1     Running   0          7s    10.128.41.190   node34.st   <none>           <none>
nginx-cqwk6   1/1     Running   0          8s    10.128.107.106  node21.st   <none>           <none>
nginx-cvc9c   1/1     Running   0          8s    10.128.5.233    node58.st   <none>           <none>
nginx-czzg8   1/1     Running   0          8s    10.128.9.173    node10.st   <none>           <none>
nginx-f7vzp   1/1     Running   0          8s    10.128.174.147  node35.st   <none>           <none>
nginx-fp7k6   1/1     Running   0          7s    10.128.125.185  node49.st   <none>           <none>
nginx-fpwdc   1/1     Running   0          7s    10.128.234.29   node12.st   <none>           <none>
nginx-g55j6   1/1     Running   0          8s    10.128.23.181   node57.st   <none>           <none>
nginx-glnx2   1/1     Running   0          7s    10.128.126.151  node52.st   <none>           <none>
nginx-gs8r2   1/1     Running   0          7s    10.128.207.234  node59.st   <none>           <none>
...
```

ITMOre than a UNIVERSITY

DaemonSet

```
kubectl describe ds nginx
Name:           nginx
Selector:       app=nginx-daemonset,lecture=k8
Node-Selector:  <none>
Labels:         lecture=k8s
Annotations:    deprecated.daemonset.template.generation:
Desired Number of Nodes Scheduled: 5
Current Number of Nodes Scheduled: 5
Number of Nodes Scheduled with Up-to-date Pods: 5
Number of Nodes Scheduled with Available Pods: 5
Number of Nodes Misscheduled:
Pods Status:  53 Running / 0 Waiting / 0 Succeeded / 0 Faile
Pod Template:
  Labels:  app=nginx-daemonset
           lecture=k8s
  Containers:
   nginx:
    Image:        nginx:1.1
    Port:         <none>
    Host Port:    <none>
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Events:
...
```

Job

ITMO UNIVERSITY

```yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: for-loop
  labels:
    lecture: k8s
spec:
  backoffLimit: 4
  template:
    metadata:
      labels:
        app: for-loop
        lecture: k8s
    spec:
      restartPolicy: Never   # OnFailure
      containers:
      - name: for-loop
        image: nginx:1.17
        command: ["sh", "-c",  "for i in `seq 1 5`; do echo $i; sleep 1; done"]
```

ITMO more than a UNIVERSITY

Job

```
kubectl apply -f nginx-job.yaml
job.batch/for-loop created

kubectl get pods -o=wide
NAME             READY   STATUS      RESTARTS   AGE
for-loop-nkmmj   0/1     Completed   0          10s

kubectl get jobs
NAME       COMPLETIONS   DURATION   AGE
for-loop   1/1           8s         34s

kubectl describe job for-loop
Name:             for-loop
Namespace:        default
Selector:         controller-uid=2a495239-f857-4153-83e4-d751b5e8a2b1
Labels:           lecture=k8s
Annotations:      <none>
Parallelism:      1
Completions:      1
Start Time:       Mon, 29 Nov 2021 05:32:21 +0300
Completed At:     Mon, 29 Nov 2021 05:32:29 +0300
Duration:         8s
Pods Statuses:    0 Running / 1 Succeeded / 0 Failed
...
Events:
  Type    Reason            Age   From            Message
  ----    ------            ----  ----            -------
  Normal  SuccessfulCreate  43s   job-controller  Created pod: for-loop-nkmmj
  Normal  Completed         36s   job-controller  Job completed
```

Thank you for the attention!

alipoov.nb@gmail.com
se.teryoshkin@gmail.com
www.ifmo.ru

ITMO more than a UNIVERSITY