Kubernetes & Akira

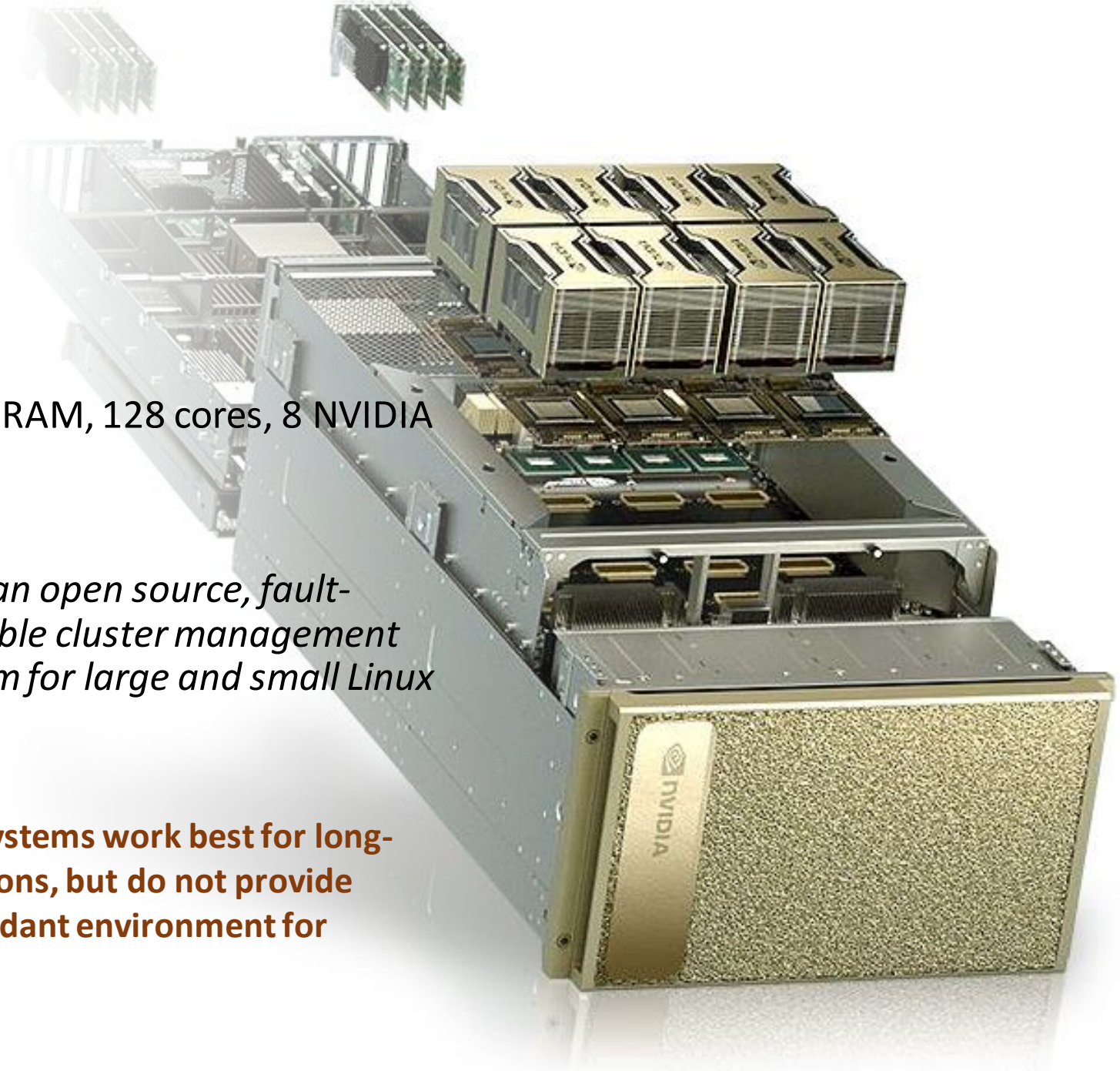
Piotr Piątyszek

28.03.2022

Infrastructure

Overview of servers available to lab members.

Eden



- 4 x DGX A100 = 4 x (1 TiB RAM, 128 cores, 8 NVIDIA A100 GPU)
- 1 PiB network storage
- Runs on Slurm - *Slurm is an open source, fault-tolerant, and highly scalable cluster management and **job scheduling** system for large and small Linux clusters*

Job scheduling systems work best for long-running calculations, but do not provide stable and redundant environment for services

Virtual Machines

- Faculty operates virtual machines hypervisor.
- Each VM have only ~2GB RAM, 20GB storage, 1 vCPU
- Configuration requires a lot of emails to Marcin Borkowski

We need a more powerful and redundant backend for most services.

MI² Infrastructure

- Simba – 1TiB Ram, 51 TiB HDD storage, 48 cores = 96 threads, 1TiB M.2 storage (simba.mini.pw.edu.pl)
- Bambi – 251 GiB Ram, ~26TiB HDD storage, 24 cores = 48 threads (bambi.mini.pw.edu.pl)
- Tarzan - 1TiB Ram, 51 TiB storage, 48 cores = 96 threads, NVIDIA A30 24GB

  In progress of ordering

We need a platform to orchestrate.

Containerized workloads

Docker

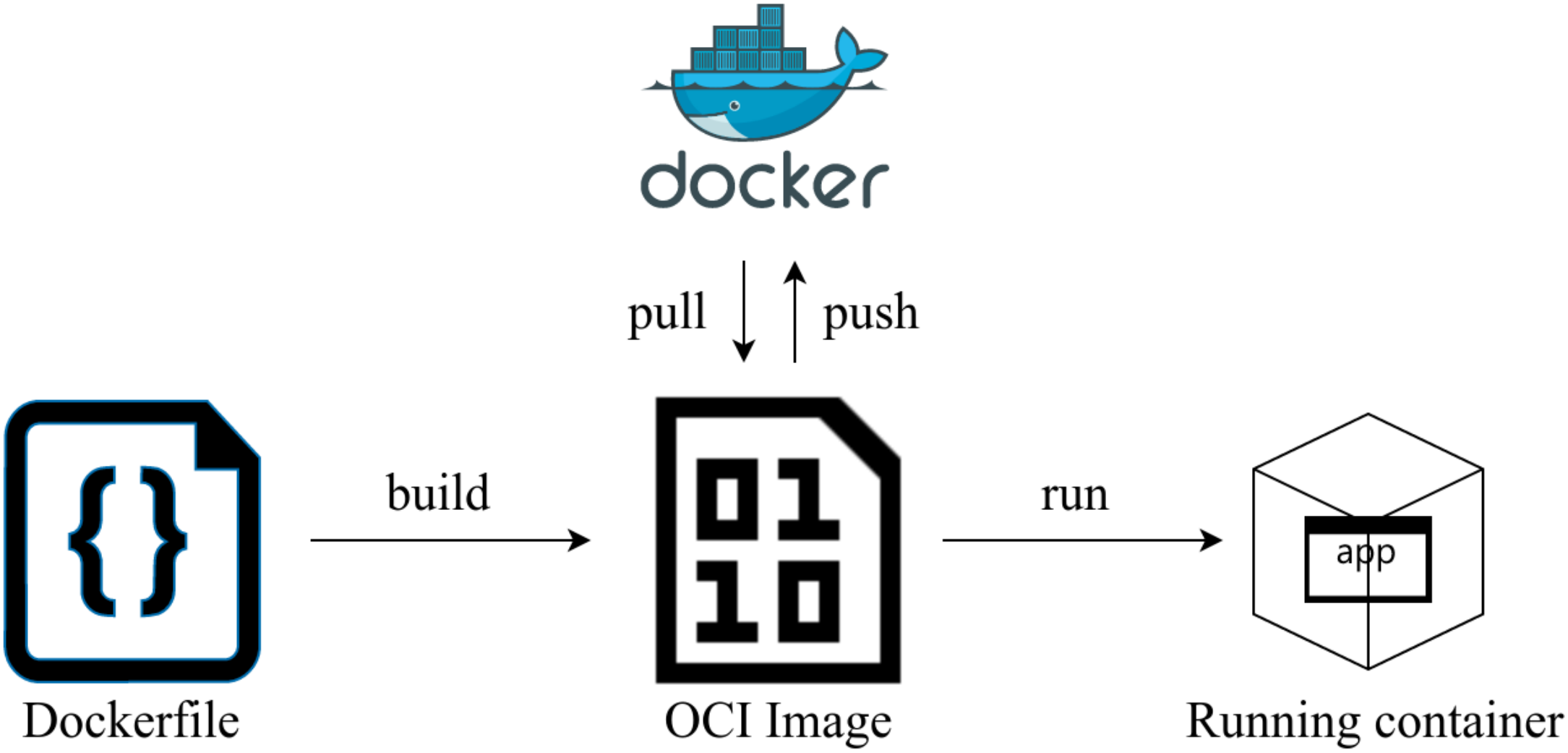
- Simple and easy if you have one computer and root account
- Alone is not suitable for multiuser clusters
- Provide universal mechanism of building images using Dockerfiles

```
FROM rocker/r-ver:4.1.2

RUN R -e 'install.packages(c("dplyr", "mlr3", "plumber"))'
RUN mkdir /work

COPY model.rds /work
COPY model.R /work

ENTRYPOINT ["R", "-e", "r = plumber::plumb('/work/model.R'); r$run(port = 1030, host = '0.0.0.0')"]
```


docker
pull
push
build
run
app
Dockerfile
OCI Image
Running container

Kubernetes

Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both **declarative configuration** and automation

Imperative configuration

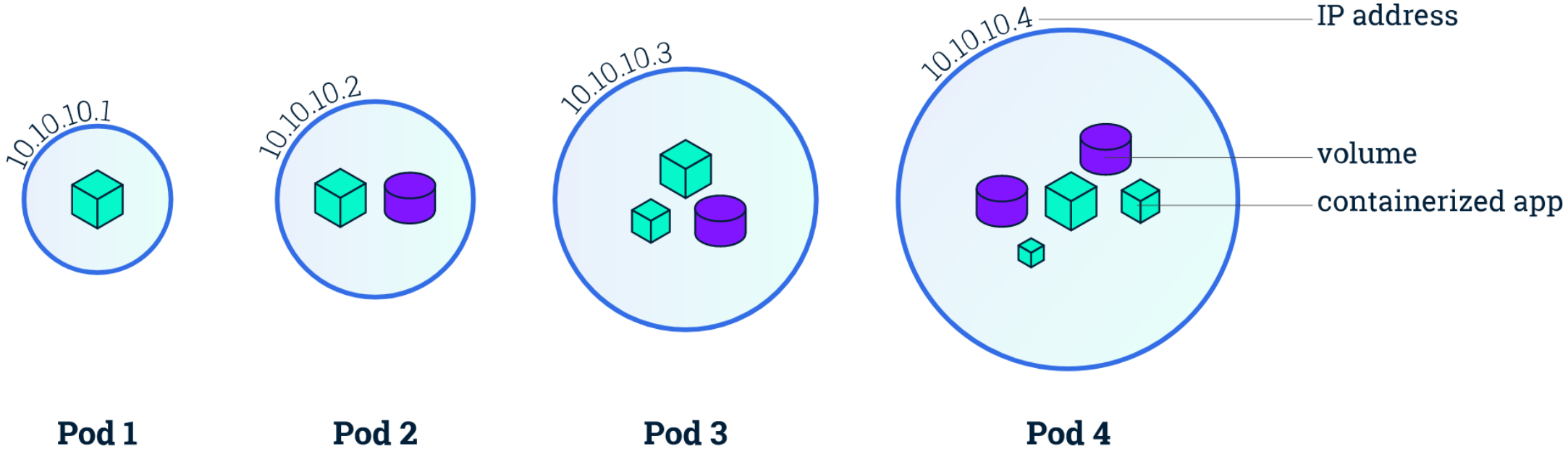
- You give explicit instructions of steps to be done
- You need to handle failure in one of steps.
- You need to observe state of the system and react to changes.

Declarative configuration

- You describe desired effect.
- System watches how it differs from current state and make actions to minimize the difference.

Resources available at Kubernetes

Pod – smallest unit of deployment
Unofficially POD="Part Of Deployment"



Pod		
metadata	name	example-pod-1
	labels	
	app	just-ubuntu
spec	image	ubuntu:latest
	limits	
	memory	2 GiB
	cpu	0.5

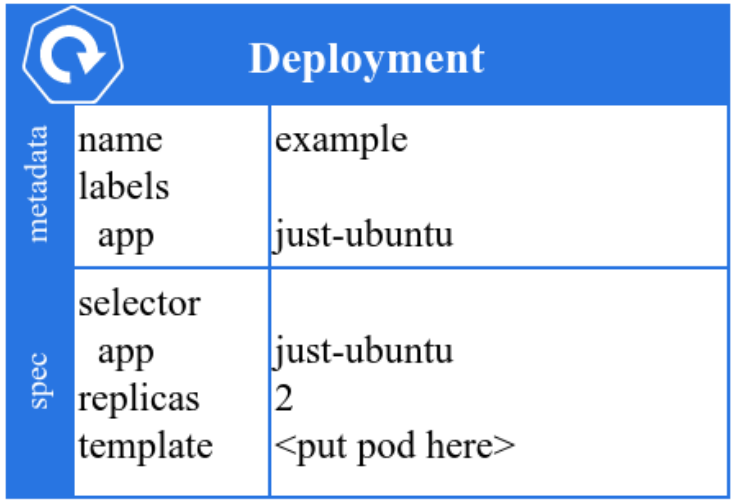
Pod is a resource in Kubernetes. Resources must have metadata with name and can have other sections like spec.

For simplicity pod in this and next schemas have only one container.

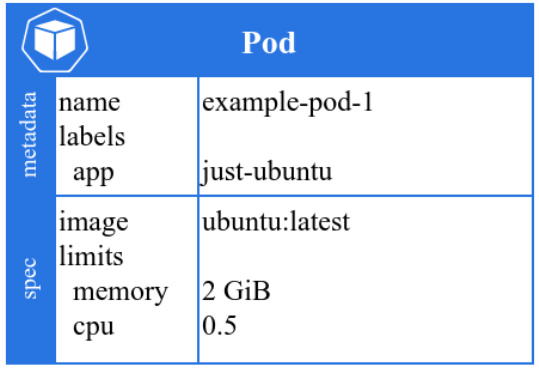
We can create multiple pods manually, but this is not recommended.

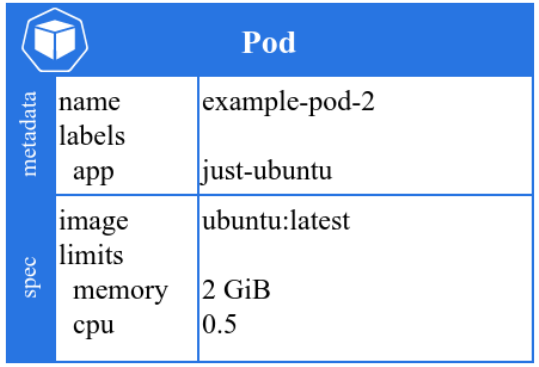
Pod		
metadata	name	example-pod-1
	labels	
	app	just-ubuntu
spec	image	ubuntu:latest
	limits	
	memory	2 GiB
	cpu	0.5

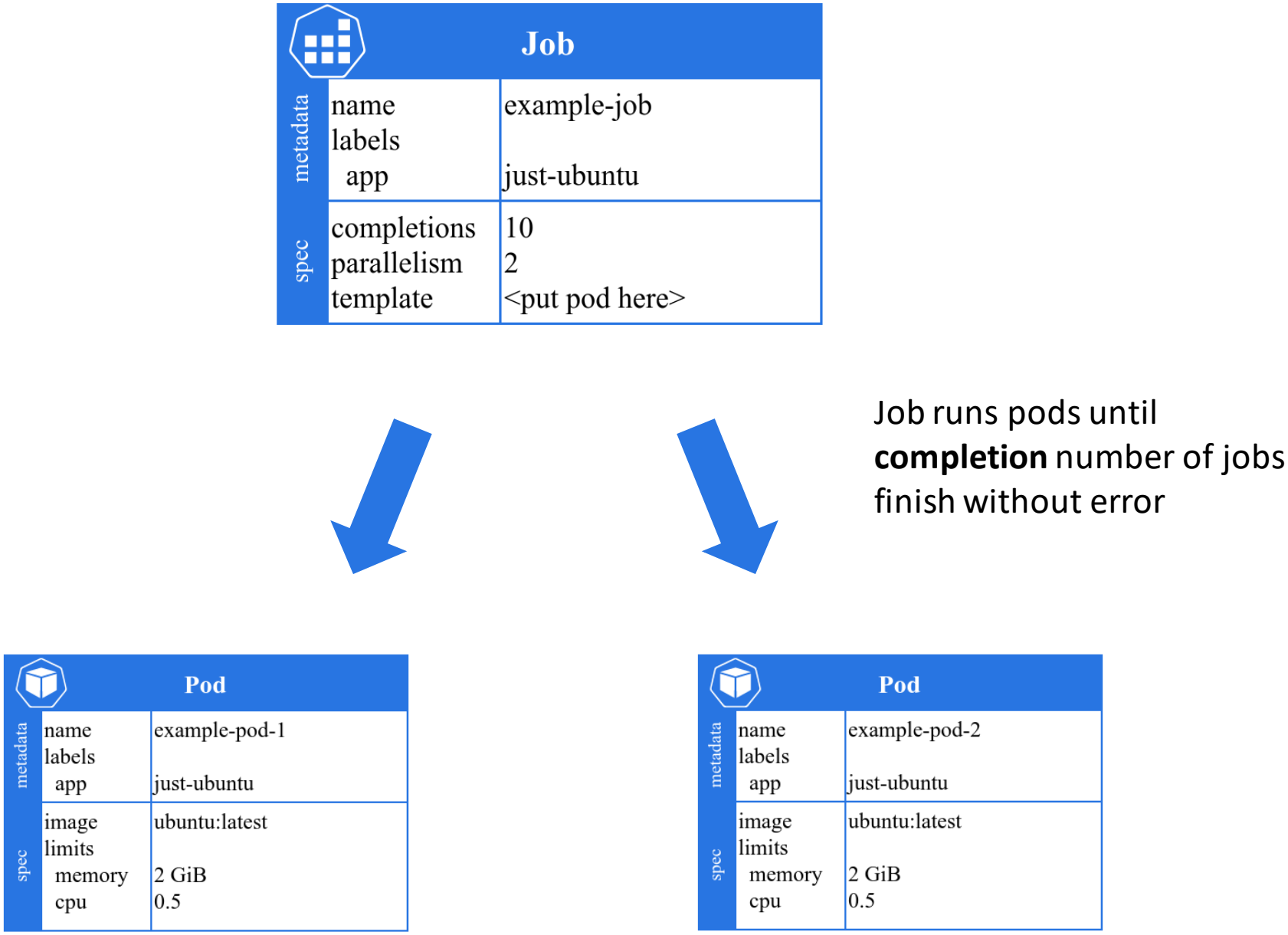
Pod		
metadata	name	example-pod-2
	labels	
	app	just-ubuntu
spec	image	ubuntu:latest
	limits	
	memory	2 GiB
	cpu	0.5

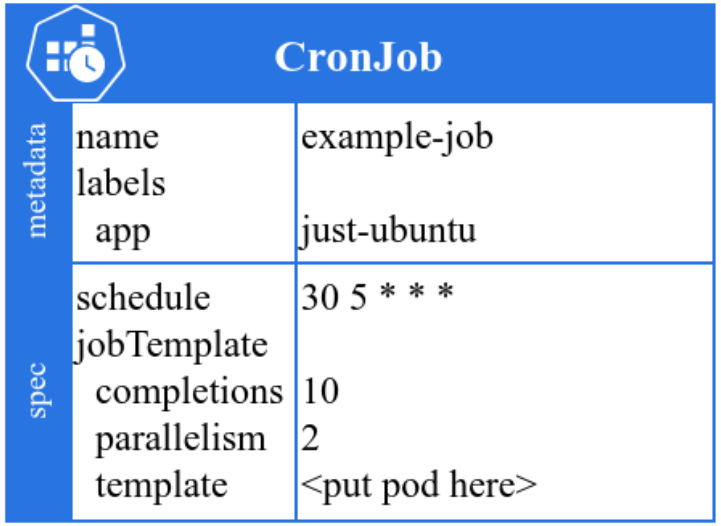


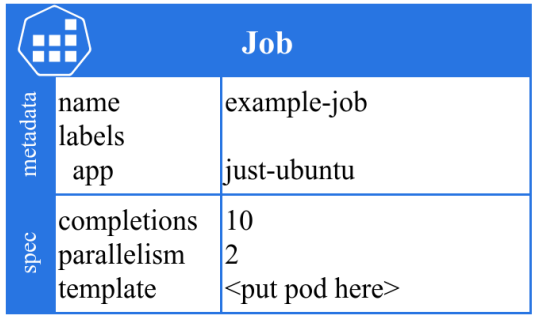
Deployment creates pods and always keep desired number of replicas running and available



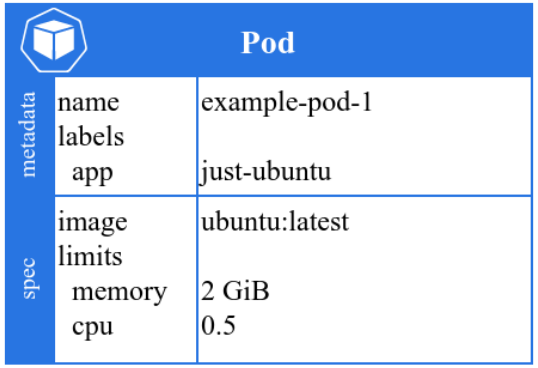


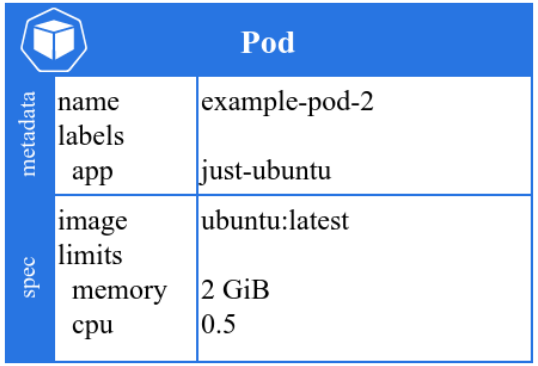

Job
metadata
name
labels
app
example-job
just-ubuntu
spec
completions
parallelism
template
10
2
<put pod here>
Job runs pods until **completion** number of jobs finish without error
Pod
metadata
name
labels
app
example-pod-1
just-ubuntu
spec
image
limits
memory
cpu
ubuntu:latest
2 GiB
0.5
Pod
metadata
name
labels
app
example-pod-2
just-ubuntu
spec
image
limits
memory
cpu
ubuntu:latest
2 GiB
0.5





CronJob runs a Job on the given schedule





Service exposes port of one or more pods to the cluster or internal network

Service		
metadata	name	example-pod-www
spec	selector	
	app	just-ubuntu
	ports	
	- protocol	TCP
	port	80
	targetPort	8080

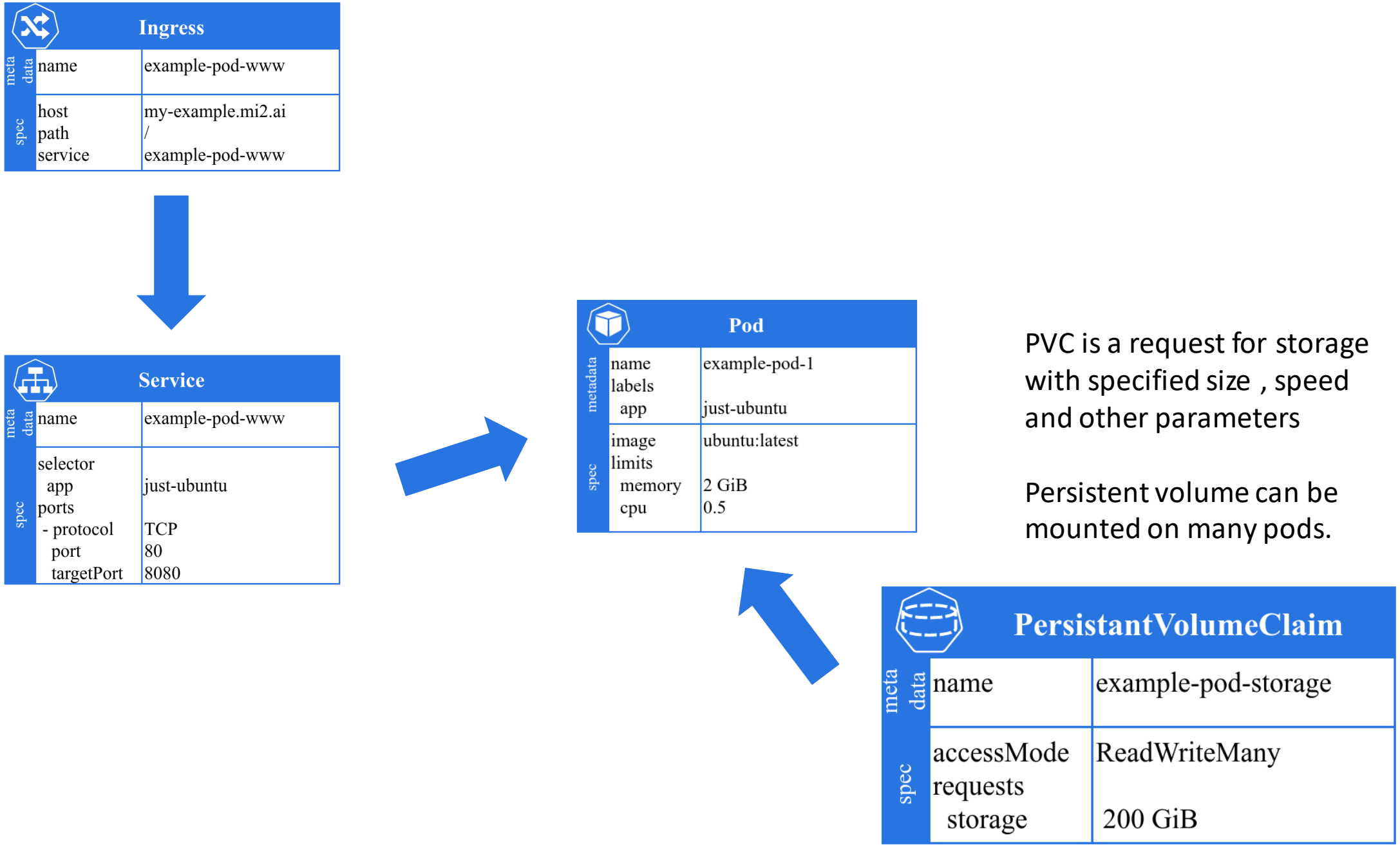
Pod		
metadata	name	example-pod-1
	labels	
	app	just-ubuntu
spec	image	ubuntu:latest
	limits	
	memory	2 GiB
	cpu	0.5

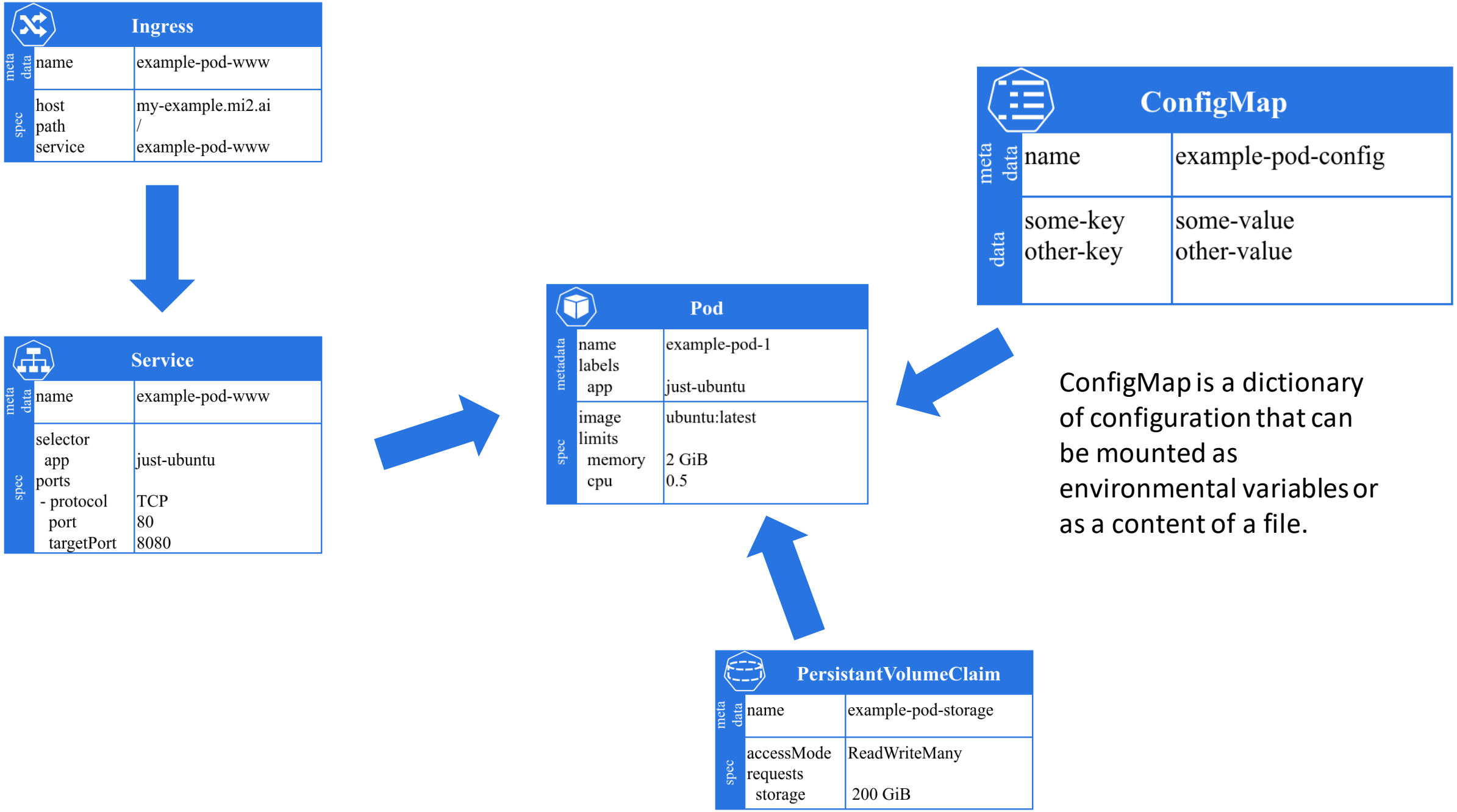
Ingress		
metadata	name	example-pod-www
spec	host path service	my-example.mi2.ai / example-pod-www

Ingress maps HTTP traffic from specified domain to your service

Service		
metadata	name	example-pod-www
spec	selector app ports - protocol port targetPort	 just-ubuntu TCP 80 8080

Pod		
metadata	name labels app	example-pod-1 just-ubuntu
spec	image limits memory cpu	ubuntu:latest 2 GiB 0.5


Ingress
metadata
name
example-pod-www
spec
host
path
service
my-example.mi2.ai
/
example-pod-www
Service
metadata
name
example-pod-www
spec
selector
app
ports
- protocol
port
targetPort
just-ubuntu
TCP
80
8080
Pod
metadata
name
labels
app
example-pod-1
just-ubuntu
spec
image
limits
memory
cpu
ubuntu:latest
2 GiB
0.5
PVC is a request for storage with specified size , speed and other parameters
Persistent volume can be mounted on many pods.
PersistantVolumeClaim
metadata
name
example-pod-storage
spec
accessMode
requests
storage
ReadWriteMany
200 GiB


Ingress
metadata
name
example-pod-www
spec
host
path
service
my-example.mi2.ai
/
example-pod-www
Service
metadata
name
example-pod-www
spec
selector
app
ports
- protocol
port
targetPort
just-ubuntu
TCP
80
8080
Pod
metadata
name
labels
app
example-pod-1
just-ubuntu
spec
image
limits
memory
cpu
ubuntu:latest
2 GiB
0.5
ConfigMap
metadata
name
example-pod-config
data
some-key
other-key
some-value
other-value
PersistantVolumeClaim
metadata
name
example-pod-storage
spec
accessMode
requests
storage
ReadWriteMany
200 GiB
ConfigMap is a dictionary of configuration that can be mounted as environmental variables or as a content of a file.

Secrets are very similar to ConfigMap, but values are set in base64 and they can be encrypted.

Ingress

metadata	name	example-pod-www
spec	host path service	my-example.mi2.ai / example-pod-www

Service

metadata	name	example-pod-www
spec	selector app ports - protocol port targetPort	 just-ubuntu TCP 80 8080

Pod

metadata	name labels app	example-pod-1 just-ubuntu
spec	image limits memory cpu	ubuntu:latest 2 GiB 0.5

Secret

metadata	name	example-pod-secret
data	db-user db-password	base64(username) base64(password)

ConfigMap

metadata	name	example-pod-config
data	some-key other-key	some-value other-value

PersistantVolumeClaim

metadata	name	example-pod-storage
spec	accessMode requests storage	ReadWriteMany 200 GiB

Namespace

Pod

metadata	name	example-pod-1
	labels	
	app	just-ubuntu
spec	image	ubuntu:latest
	limits	
	memory	2 GiB
	cpu	0.5

Service

metadata	name	example-pod-www
spec	selector	
	app	just-ubuntu
	ports	
	- protocol	TCP
	port	80
	targetPort	8080

ConfigMap

metadata	name	example-pod-config
data	some-key	some-value
	other-key	other-value

PersistantVolumeClaim

metadata	name	example-pod-storage
spec	accessMode	ReadWriteMany
	requests	
	storage	200 GiB

Namespace gathers resources that can access and use each other.

Namespaces are targets of role bindings. The user is allowed to do action on all resources of a given type or on none of them in the namespace.

Managing resources

Verbs in API

- Create
- Delete
- Get
- List
- Watch
- Update
- Patch

Kubectl

- Command line tool for communication with Kubernetes API
- Most important commands
  - kubectl apply –f <path to manifest>
    Creates or patches resources defined in manifest
  - kubectl delete <resource> <name>
    Removes resource of given name and type
  - kubectl list <resource>
    List resources of given type
  - kubectl get <resource> <name> -o yaml
    Returns manifest of resource with given name and type

K9s – interactive kubectl

```
Context:     minikube                    <?>      Help        <0> all
Cluster:     minikube                    <ctrl-d> Delete      <1> kube-system
User:        minikube                    <d>      Describe    <2> default
K9s Version: 0.1.6                       <e>      Edit
K8s Version: v1.13.2                     <l>      Logs
CPU:         10%(-)                      <s>      Shell
MEM:         20%(+)                      <v>      View

                                        Pods(all)[12]
NAMESPACE   NAME                          READY STATUS  RESTARTS CPU     MEM     IP            NODE          QOS        AGE
default     nginx-6988c9989f-wwz6d        1/1   Running 0                        172.17.0.6    192.168.64.83 Guaranteed 87s
kube-system coredns-86c58d9df4-dkhf2      1/1   Running 0        3m      8Mi     172.17.0.3    192.168.64.83 Burstable  17h
kube-system coredns-86c58d9df4-jt79s      1/1   Running 0        2m(-)   8Mi     172.17.0.2    192.168.64.83 Burstable  17h
kube-system etcd-minikube                 1/1   Running 0        24m(-)  53Mi    192.168.64.83 192.168.64.83 BestEffort 17h
kube-system kube-addon-manager-minikube   1/1   Running 0        7m(+)   17Mi(-) 192.168.64.83 192.168.64.83 Burstable  17h
kube-system kube-apiserver-minikube       1/1   Running 0        47m(-)  383Mi   192.168.64.83 192.168.64.83 Burstable  17h
kube-system kube-controller-manager-minikube 1/1 Running 0        49m(-)  55Mi(+) 192.168.64.83 192.168.64.83 Burstable  17h
kube-system kube-proxy-pjh2p              1/1   Running 0        4m(-)   10Mi    192.168.64.83 192.168.64.83 BestEffort 17h
kube-system kube-scheduler-minikube       1/1   Running 0        15m(-)  12Mi    192.168.64.83 192.168.64.83 Burstable  17h
kube-system kubernetes-dashboard-ccc79bfc9-qgnck 1/1 Running 0    0m      11Mi    172.17.0.4    192.168.64.83 BestEffort 17h
kube-system metrics-server-6fc4b7bcff-hp6pm 1/1 Running 0         1m      14Mi    172.17.0.5    192.168.64.83 BestEffort 17h
kube-system storage-provisioner           1/1   Running 0        0m      13Mi    192.168.64.83 192.168.64.83 BestEffort 17h
```

Flux

- Synchronizes manifests in GitHub repository with Kubernetes
- GitOps – versioning of deployment
- Allows automated updates of images
- Secrets can be encrypted

MR2 AKIRA

Nodes

- Janusz – Virtual Machine working as a master node
- Simba – 1TiB Ram, 51 TiB HDD storage, 48 cores = 96 threads, 1TiB M.2 storage (simba.mini.pw.edu.pl)
- Bambi – 251 GiB Ram, ~26TiB HDD storage, 24 cores = 48 threads (bambi.mini.pw.edu.pl)

  Not connected yet
- Tarzan - 1TiB Ram, 51 TiB storage, 48 cores = 96 threads, NVIDIA A30 24GB

  In progress of ordering

Access Node

- Available using SSH at **akira.mini.pw.edu.pl** in MINI network
- Command **renew** generates certificates for Kubernetes API
- Configured kubectl, k9s, kaniko-builder are available

Namespaces

- Each user gets its own individual namespace
- Each namespace has connected GitHub repository for configuration
- Each project realized in lab can get one or more namespaces for common work

Services

- ClusterIP / Headless – reachable withing cluster
- NodePort – exposes port on all nodes ex. simba.mini.pw.edu.pl:32123
- LoadBalancer – exposes port on akira.mini.pw.edu.pl and balances traffic to all pods

Ingress

- Domains *.mi2.ai are available
- Easy password protection
- Traffic can be limited to internal MINI network
- SSL termination

Storage

- Based on local ZFS filesystem
- Volumes can be shared only from one node (local volumes)
- Available compression and deduplication mechanisms
- Record size can be chosen from 4KB to 256KB – the first is faster for small modifications and the other for large sequential writes
- Supports fast and cheap snapshots and clones (Copy on Write mechanism)

Network Policy

- Network policies are fully customizable using Calico
- By default, all traffic inside the namespace is allowed. Egress to the internet and critical services is allowed. Ingress is allowed only from Nginx to pods with label www=true.

Image registry

- Self-hosted Quay registry is available at https://quay.mi2.ai
- Each user get one account with unlimited number of repositories and storage space.
- Users can create robot accounts with limited permissions and separated credentials.

Building images

- You can build on your local computer using *docker build*
- Kaniko allows building images on cluster from access node
- GitHub Actions support *docker build* and *BuildKit*

Demo