

“Swarm & K8S”

We are one big community

By Praparn Luengphoonlap
Email: eva10409@gmail.com

“Swarm & K8S” We are one big community

Agenda

- Highlight from DockerCon Europe 2017
- Benefit of K8S in docker platform
- K8S vs Docker (Swarm) what is the difference ?
- Demo Session
 - Enable K8S feature and deploy dashboard for K8S
 - Demo: Deploy simple application with K8S and Docker Native
 - Demo: Deploy compose with K8S and Swarm

“Swarm & K8S” We are one big community



Highlight from DockerCon17 (EU)



"Swarm & K8S" We are one big community

Highlight from DockerCon17 (EU)

- “We are one big community”

WE ARE
ONE BIG
COMMUNITY



“Swarm & K8S” We are one big community

Highlight from DockerCon17 (EU)

- Official support orchestrator both K8S and Swarm

Docker with Swarm and Kubernetes

1 →

The best enterprise
container security and
management

Docker Enterprise Edition

2 ←

The best container
development workflow

3 →

Native Kubernetes
integration provides full
ecosystem
compatibility

Docker Community Edition



kubernetes



containerd

2 ←

4 ←

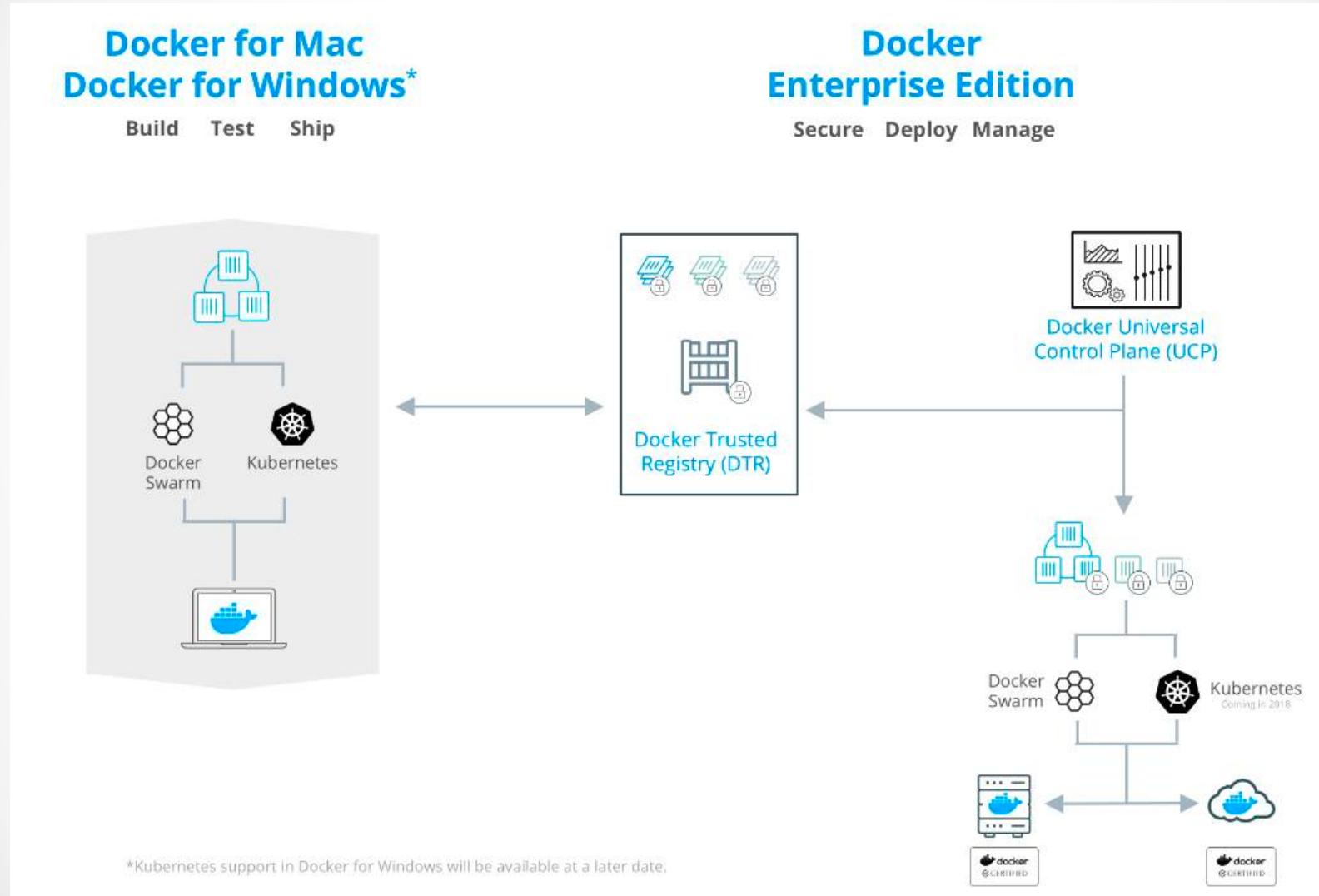
Industry-standard
container runtime



"Swarm & K8S" We are one big community



Highlight from DockerCon17 (EU)



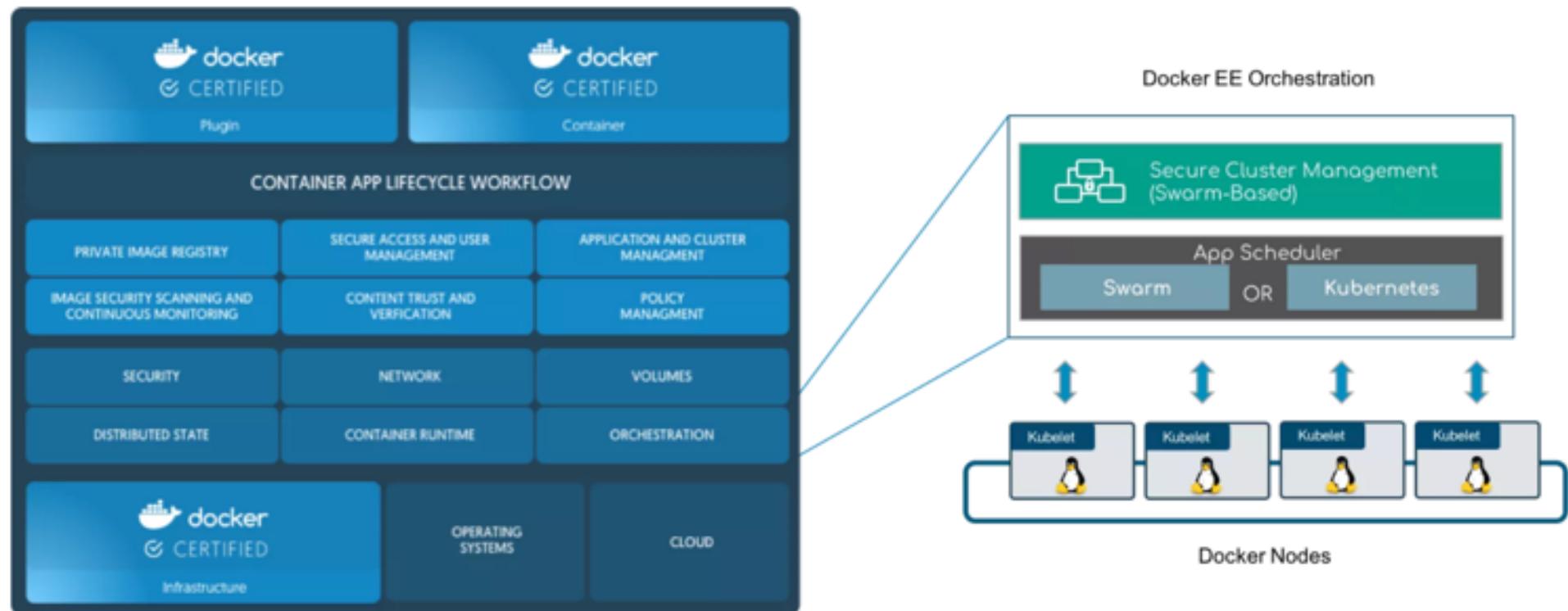
“Swarm & K8S” We are one big community

Highlight from DockerCon17 (EU)

- The **difference concept** of Docker/K8S
 - Docker Native/Swarm Mode:
 - Simplify (Single command for initial/manage/maintain)
 - Provide all resource within suite (overlay network, key value etc) by default
 - Fastest deployment
 - K8S (Kubernetes):
 - Production grade orchestrator (Customizable)
 - Customize by design (network plugin, storage etc)
 - Complex configuration support for handle application

Highlight from DockerCon17 (EU)

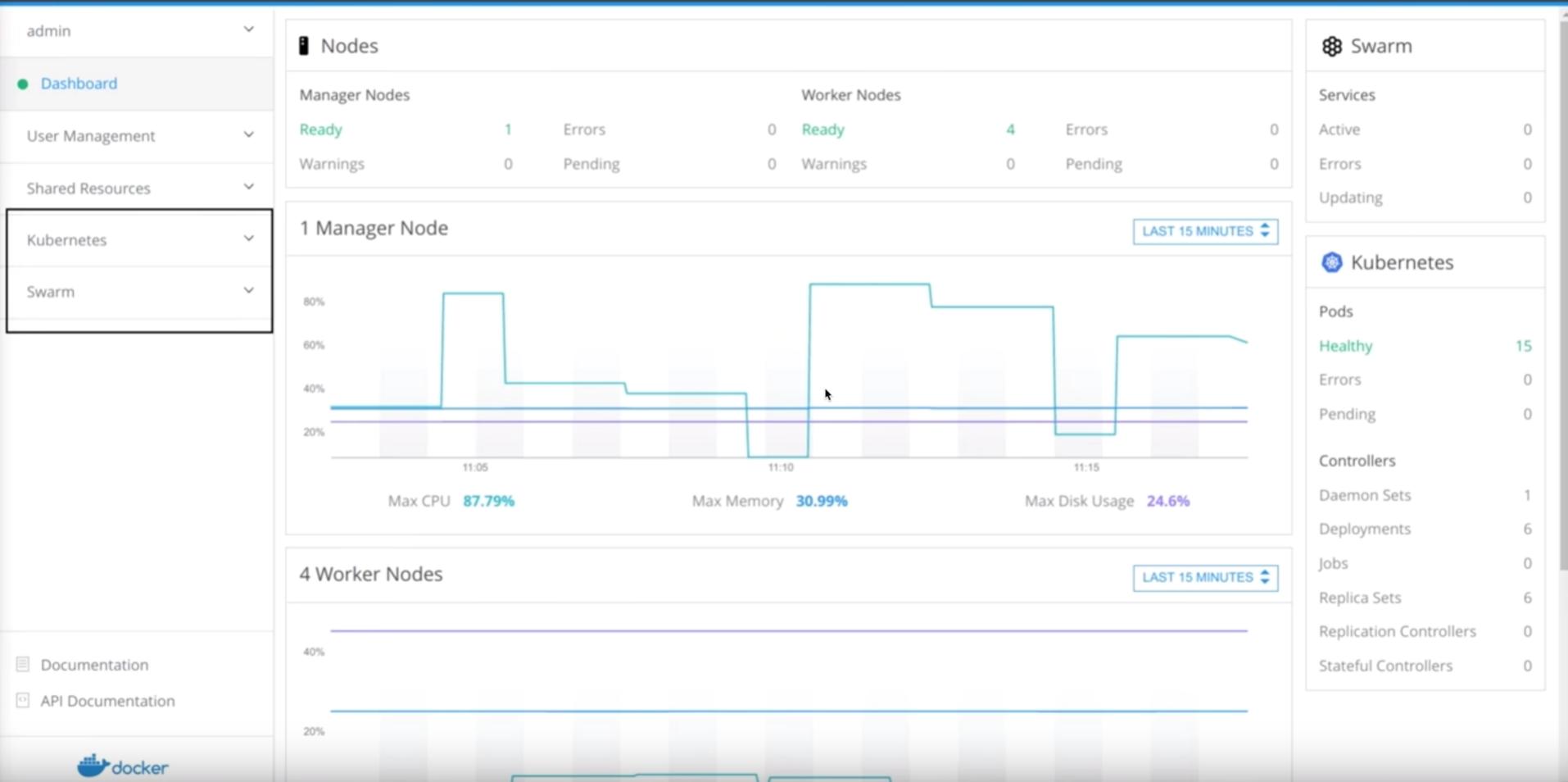
- Platform of docker to support K8S
- Docker Enterprise Edition



"Swarm & K8S" We are one big community

Highlight from DockerCon17 (EU)

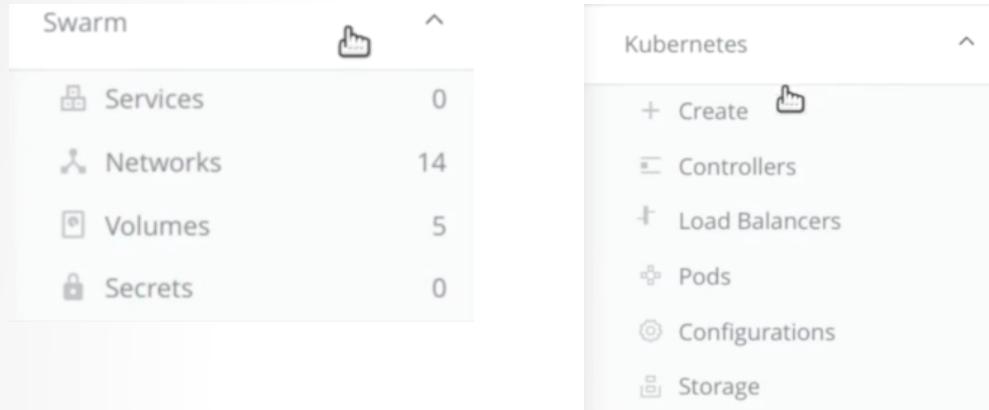
- Docker Enterprise Edition (EE)



“Swarm & K8S” We are one big community

Highlight from DockerCon17 (EU)

- Docker Enterprise Edition (EE)



The screenshot shows two side-by-side dashboards. The left dashboard is for 'Swarm' and lists:

- Services: 0
- Networks: 14
- Volumes: 5
- Secrets: 0

The right dashboard is for 'Kubernetes' and lists:

- + Create
- Controllers
- + Load Balancers
- + Pods
- @@ Configurations
- Storage

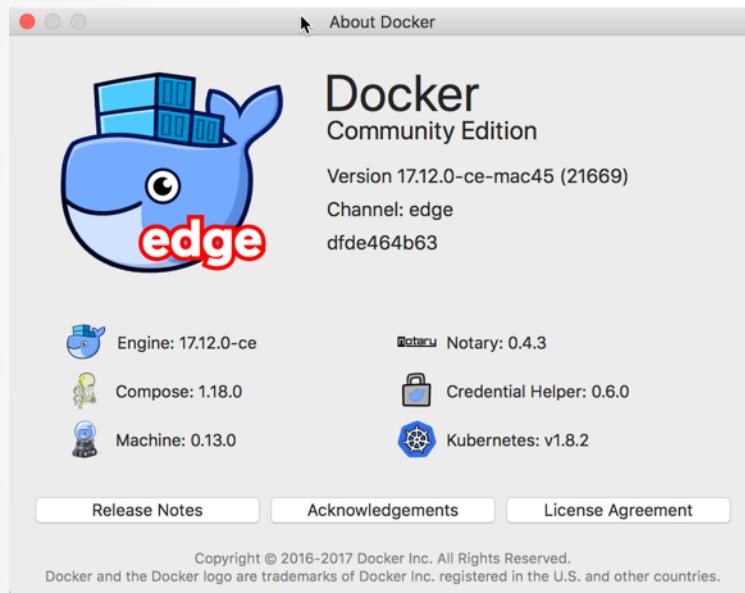
Below the dashboards is a terminal window with the following command outputs:

```
~/demo/1 docker node ls
ID           HOSTNAME        STATUS  AVAILABILITY  MANAGER STATUS
6q3c3s9jw9vq3hfd68a60npf1  dockercon1-ubuntu-3  Ready   Active
kb17pilk4ey3qdsim41finjer dockercon1-ubuntu-4  Ready   Active
uvwxyz5da05vv34jktdqovpsmxw dockercon1-ubuntu-5  Ready   Active
uvrhlku71k23081d6lfhrhqja  dockercon1-ubuntu-1  Ready   Active
xjpjsdrc88qew4kkvtpwrvooff *  dockercon1-ubuntu-0  Ready   Active
xqjltu9980fhagsystl0in1l2    dockercon1-ubuntu-2  Ready   Active
~/demo/1 kubectl get nodes
NAME            STATUS  AGE   VERSION
6q3c3s9jw9vq3hfd68a60npf1  Ready   1d    v1.7.6
kb17pilk4ey3qdsim41finjer Ready   1d    v1.7.6
uvwxyz5da05vv34jktdqovpsmxw Ready   1d    v1.7.6
uvrhlku71k23081d6lfhrhqja  Ready   1d    v1.7.6
xjpjsdrc88qew4kkvtpwrvooff Ready   1d    v1.7.6
xqjltu9980fhagsystl0in1l2    Ready   1d    v1.7.6
```

"Swarm & K8S" We are one big community

Highlight from DockerCon17 (EU)

- Docker Community Edition (CE)
 - Docker for Windows
 - Docker for MAC (**Available Now with Edge Edition !!!**)



“Swarm & K8S” We are one big community

Highlight from DockerCon17 (EU)

- Docker Community Edition (CE)
 - Docker for Windows
 - Docker for MAC (**Available Now with Edge Edition !!!**)

The screenshot shows the Kubernetes dashboard interface. On the left, there's a sidebar with navigation links for Cluster, Namespaces, Nodes (which is selected), Persistent Volumes, Roles, Storage Classes, Namespace (default), Overview, Workloads, Cron Jobs, Daemon Sets, Deployments, Jobs, Pods, and Replica Sets. The main area is titled 'Nodes' and lists a single node named 'docker-for-desktop'. The node details are as follows:

Name	Labels	Ready	CPU requests (cores)	CPU limits (cores)	Memory requests (bytes)	Memory limits (bytes)	Age
docker-for-desktop	beta.kubernetes.io/arch: am, beta.kubernetes.io/os: linux, kubernetes.io/hostname: do, node-role.kubernetes.io/ma..	True	0.81 (20.25%)	0 (0.00%)	110 Mi (5.50%)	170 Mi (8.51%)	46 minutes

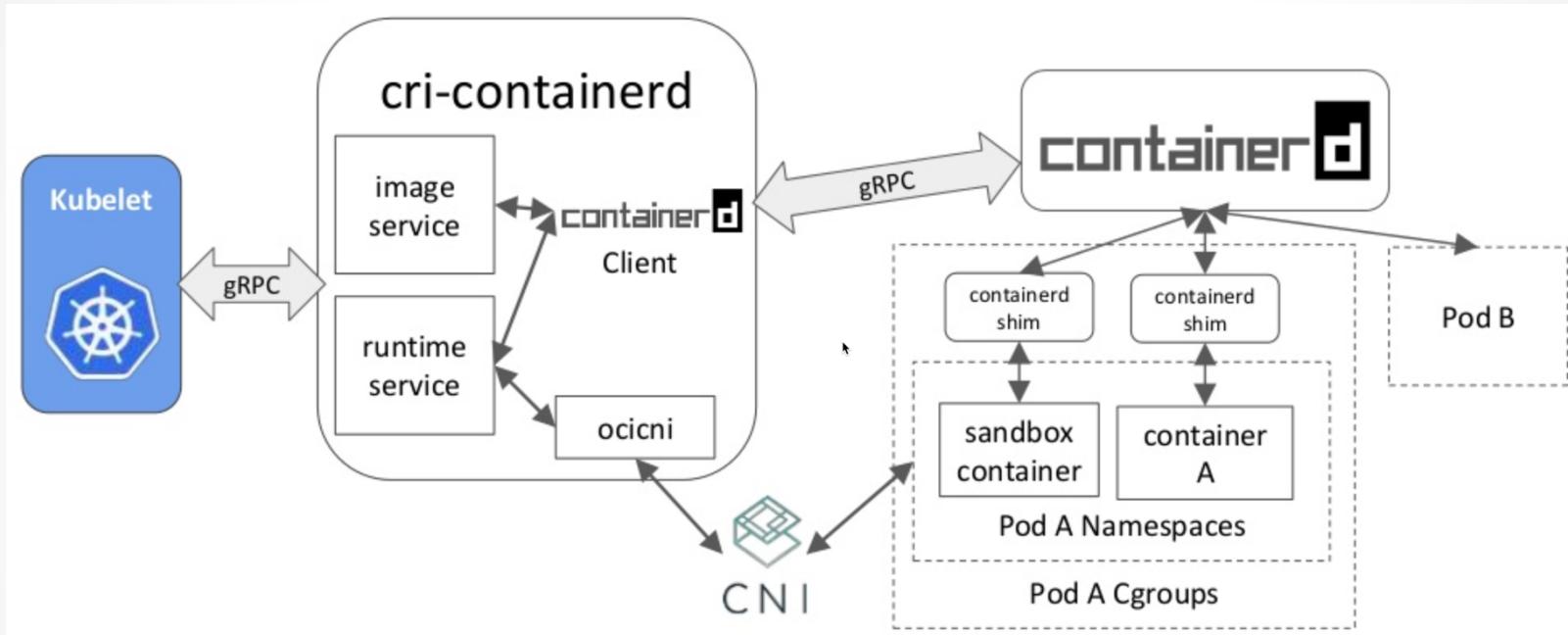
Below the nodes table, there's a terminal window showing the output of a 'kubectl port-forward' command. The terminal title is 'praparnlueangphoonlap — Terminal MAC Pro — kubectl port-forward kubernetes-dashboard-7798c48646-hvpdb 8443:8443 --names...'. The output shows:

```
No resources found.  
praparn-MacBook-Pro% kubectl get pods --namespace=kube-system  
NAME READY STATUS RESTARTS AGE  
etcd-docker-for-desktop 1/1 Running 0 45m  
kube-apiserver-docker-for-desktop 1/1 Running 0 45m  
kube-controller-manager-docker-for-desktop 1/1 Running 0 44m  
kube-dns-545bc4bfd4-vcbrx 3/3 Running 0 44m  
kube-proxy-569cd 1/1 Running 0 44m  
kube-scheduler-docker-for-desktop 1/1 Running 0 44m  
kubernetes-dashboard-7798c48646-hvpdb 1/1 Running 0 2m  
praparn-MacBook-Pro% kubectl port-forward kubernetes-dashboard-7798c48646-hvpdb 8443:8443 --namespace=kube-system  
Forwarding from 127.0.0.1:8443 -> 8443
```

"Swarm & K8S" We are one big community

Highlight from DockerCon17 (EU)

- Moby Project



kubernetes

“Swarm & K8S” We are one big community

Benefit of K8S in docker platform

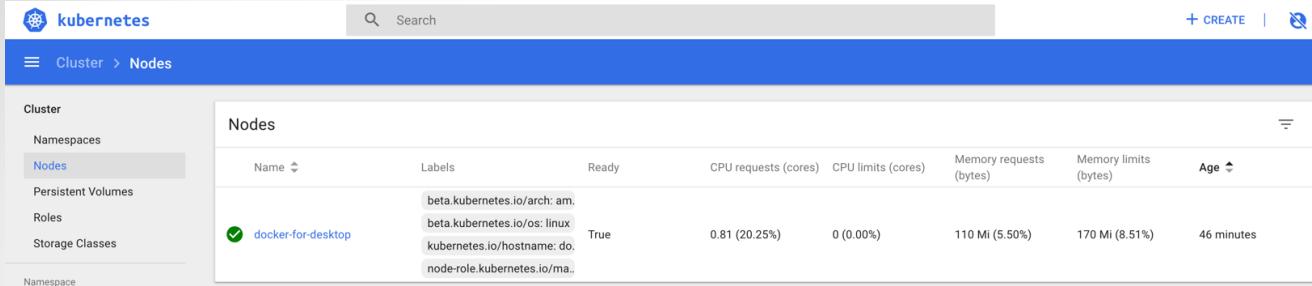


“Swarm & K8S” We are one big community

Benefit of K8S in docker platform

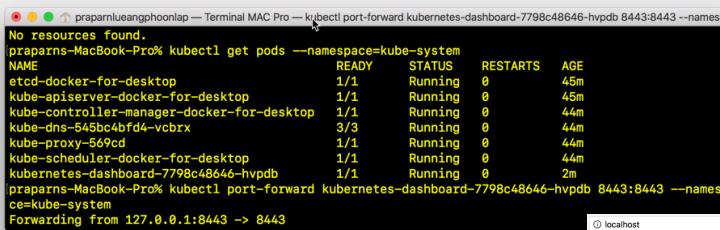
- Reduce complexity of K8S setup in this platform (Initial Farm, HA Setup etc)
- Utilization host resource from mix workload for Swarm and K8S (Docker EE)
- Developer feel free for testing both Swarm/K8S environment on their machine (Docker CE)
- Multiple operate support in same farm
 - Docker native command (docker run, docker ps etc)
 - KuberneTEST native command (kubectl create, kubectl get pods etc)
 - Compose support both orchestrator
- Big community of sharing in container world

Benefit of K8S in docker platform



```
praparns-MacBook-Pro% kubectl get pods --namespace=kube-system
No resources found.
```

```
praparns-MacBook-Pro% kubectl port-forward kubernetes-dashboard-7798c48646-hvpdb 8443:8443 --namespace=kube-system
Forwarding from 127.0.0.1:8443 -> 8443
```



```
praparns-MBP% docker stack deploy --compose-file compose.example.yml k8stack
Stack k8stack was created
Waiting for the stack to be stable and running...
- Service web has one container running
- Service words has one container running
Stack k8stack is stable and running

praparns-MBP% docker stack ls
NAME          SERVICES
k8stack        3

praparns-MBP% kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
ce-4dd0544587-lrxvr  1/1   Running   0          42s
web-564bb6fcf-21s4v  1/1   Running   0          42s
words-6bfdbd44c-4fm6  1/1   Running   0          42s
words-6bfdbd44c-j8r9  1/1   Running   0          42s
words-6bfdbd44c-rmg08 1/1   Running   0          42s
words-6bfdbd44c-wqhz  1/1   Running   0          42s
words-6bfdbd44c-wxxtf 1/1   Running   0          42s
```

```
praparns-MBP%
```

"Swarm & K8S" We are one big community



K8S vs Docker what is the difference ?



“Swarm & K8S” We are one big community

K8S vs Docker what is the difference ?

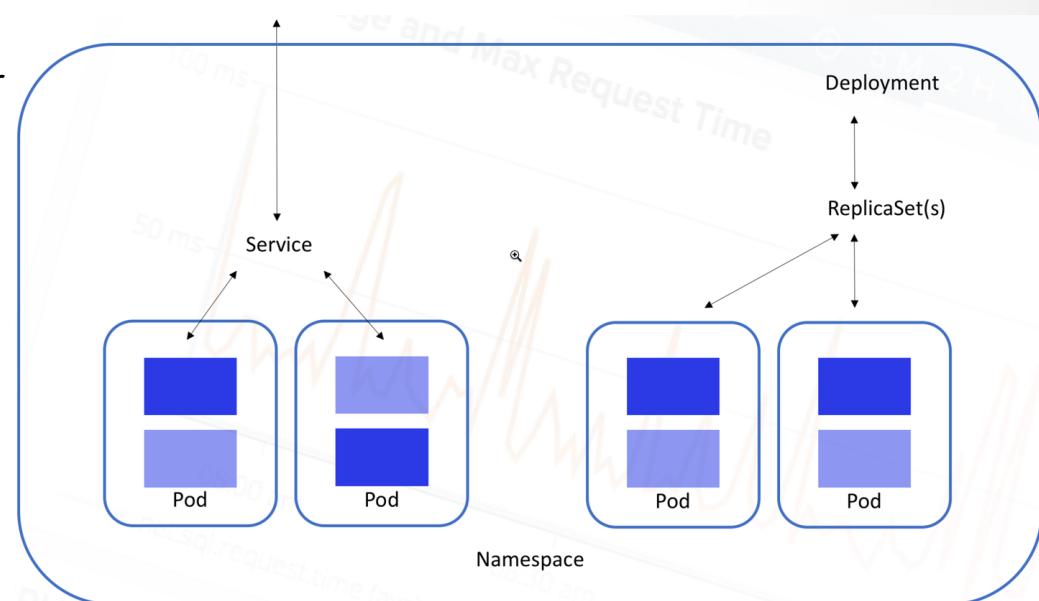
Topic	K8S	Docker/Swarm
Architecture	Open-system (Base on cluster manager "Borg" for support complex workload)	Swarm: Proprietary of Docker product, "Easy to use", "Extend capability of Docker in cluster"
Operation command	Almost operate by "YAML" file (Declarative Command)	Almost operate by "command" (Imperative Command)
Unit of Work	Pods (Pods >= Container)	Container
How to Identify Work	"Label operation"	Docker: By container name Swarm: By service/stack name
Level of workload management	Service Level: (Simple) Replication Level: (Auto healing) Deployment Level: (Auto healing + Roll Update)	Docker: N/A Swarm: Service Level (Snag with service/stack)
Auto scaling	HPA (Horizontal Pods Scaling) base on CPU	No
Health check	Liveness & Readiness (Multi option to check application health)	Service health only

"Swarm & K8S" We are one big community



K8S vs Docker what is the difference ?

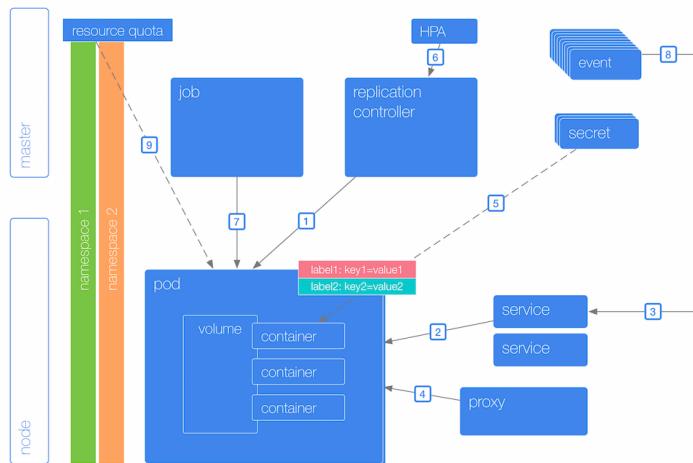
- Workload Management
 - Swarm operation
 - docker service (run/ps/stop/rm etc)
 - docker stack (deploy/ls/ps/services etc)
 - K8S Operation
 - Service
 - Replication Controller
 - Deployment + RS



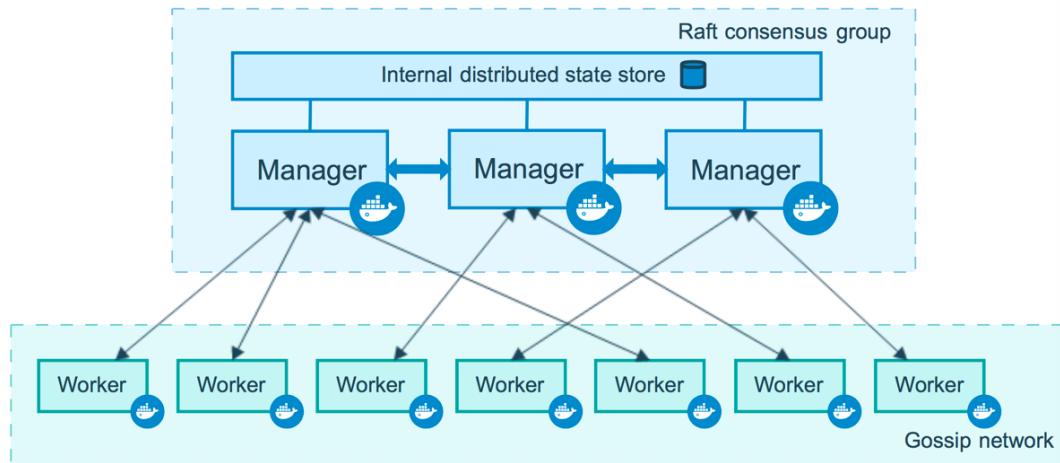
“Swarm & K8S” We are one big community

K8S vs Docker what is the difference ?

- K8S Architecture



- Docker Swarm Architcture



“Swarm & K8S” We are one big community

K8S vs Docker what is the difference ?

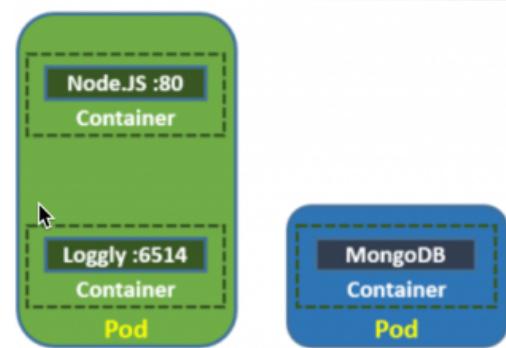
- Docker Command
 - “`docker container run -dt --name webtest -p 5000:5000 labdocker/cluster:webserviceelite`”
- K8S Pods&Service (YAML File)

```
1 apiVersion: "v1"
2 kind: Pod
3 metadata:
4   name: webtest
5   labels:
6     name: web
7     owner: Praparn_L
8     version: "1.0"
9     module: WebServer
10    environment: development
11  spec:
12    containers:
13      - name: webtest
14        image: labdocker/cluster:webserviceelite
15        ports:
16          - containerPort: 5000
17            protocol: TCP
```

```
1 apiVersion: v1
2 kind: Service
3 metadata:
4   name: webtest
5   labels:
6     name: web
7     owner: Praparn_L
8     version: "1.0"
9     module: WebServer
10    environment: development
11  spec:
12    selector:
13      name: web
14      owner: Praparn_L
15      version: "1.0"
16      module: WebServer
17      environment: development
18
19    type: NodePort
20    ports:
21      - port: 5000
22        name: http
23        targetPort: 5000
24        protocol: TCP
25
```

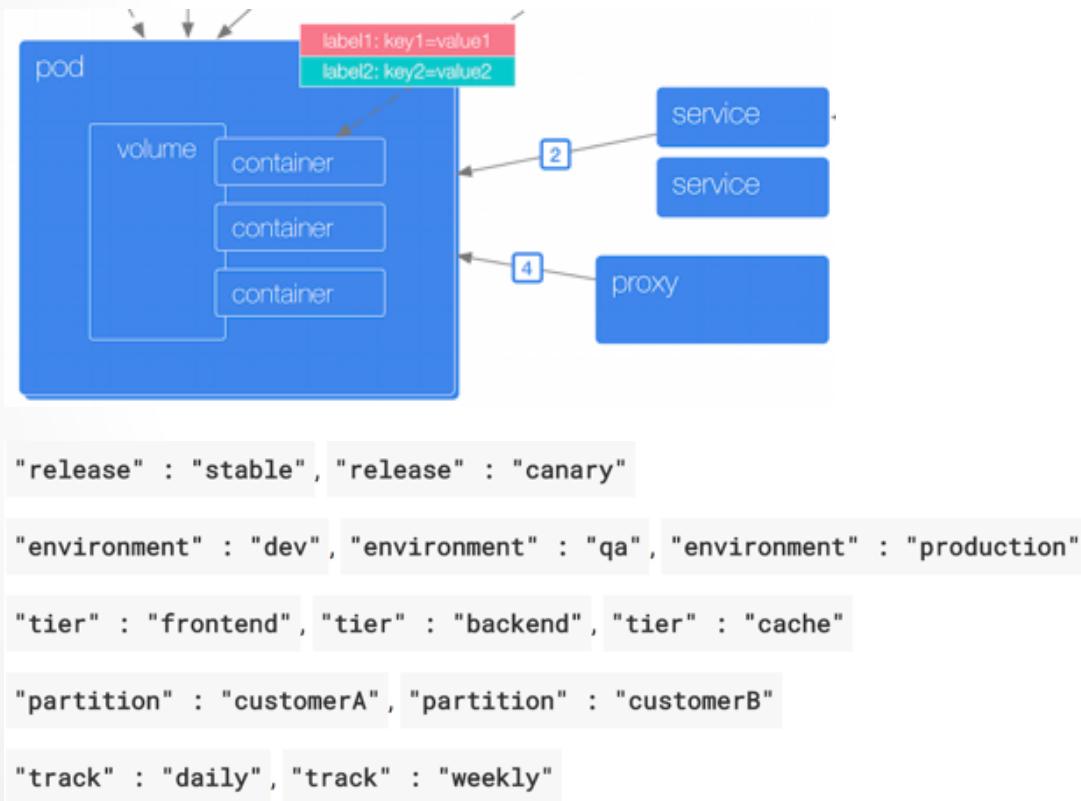
K8S vs Docker what is the difference ?

- Unit of Work (Pods vs Container)
 - All container on same Pods will share:
 - Process ID (PID)
 - Network access (Communicate to each other via “localhost”)
 - Internal Process Command (IPC)
 - Unix Time-Sharing (UTS)
 - Hostname
 - IP Address/Ports
 - Use Case for Multiple Pods:
 - Apache (1 Container) +Tomcat (1 Container)
 - Apache(1 Container) + PHP (1 Container)
 - Nginx (Cache: 1 Container) + Apache/PHP (1 Container)
 - Web Server (1 Container) + Data Volume(Cache: 1 Container)
 - Pods will can create replicas of 1000+ set on cluster system



K8S vs Docker what is the difference ?

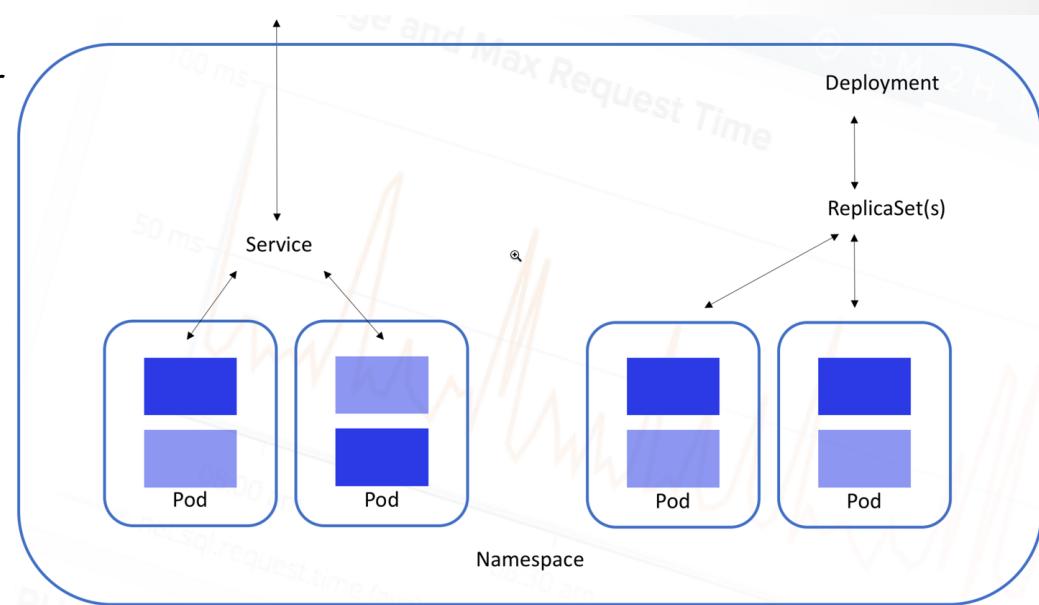
- Identify Work (Label is base advantage of K8S)
 - All component of K8S will blinding with “label”
 - Label is free-form, Customize by yourself !!!



```
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: webtest
5    labels:
6      name: web
7      owner: Praparn_L
8      version: "1.0"
9      module: WebServer
10     environment: development
11
12   spec:
13     selector:
14       name: web
15       owner: Praparn_L
16       version: "1.0"
17       module: WebServer
18       environment: development
19
20     type: NodePort
21     ports:
22       - port: 5000
23         name: http
24         targetPort: 5000
25         protocol: TCP
```

K8S vs Docker what is the difference ?

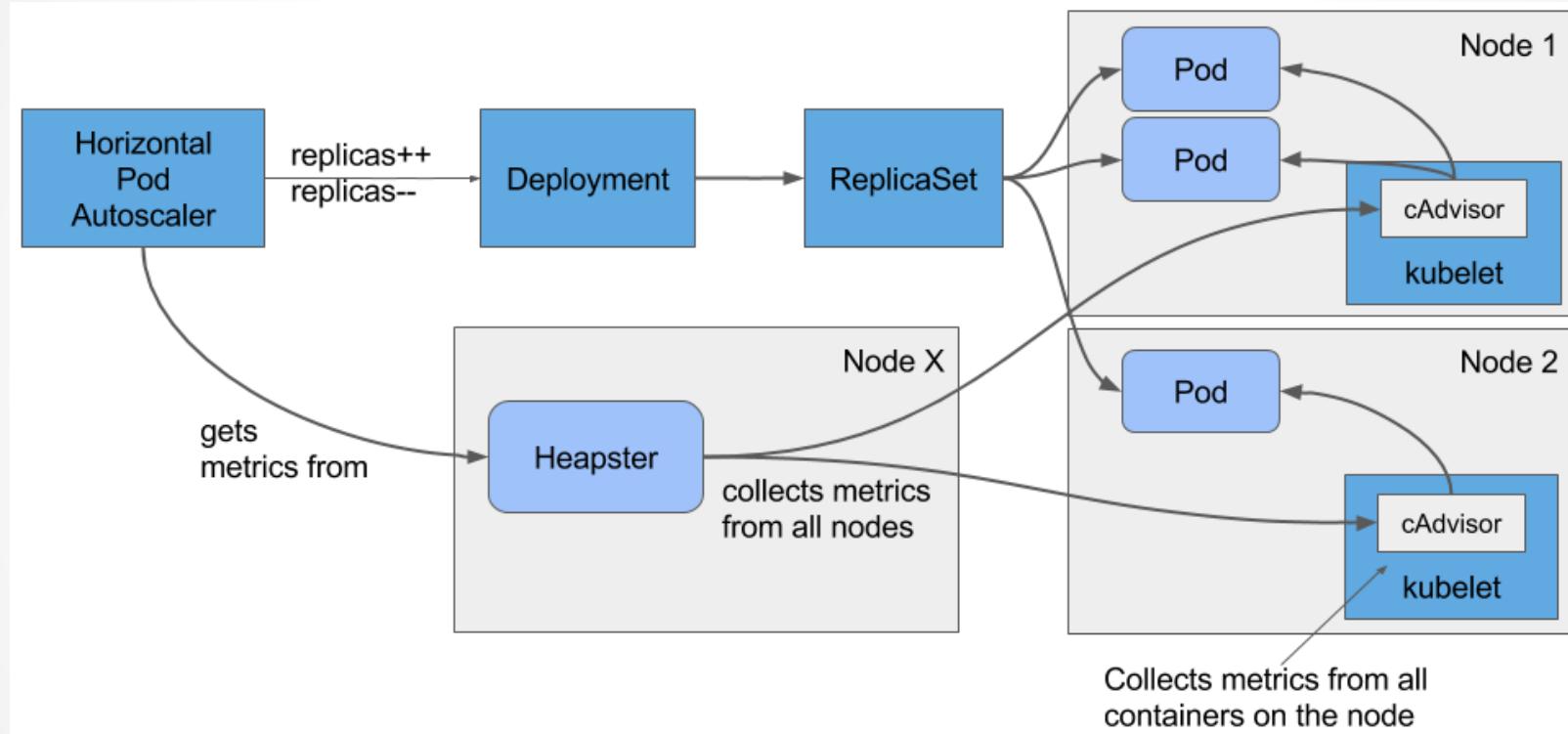
- Workload Management
 - Swarm operation (Service/Stack)
 - docker service (run/ps/stop/rm etc)
 - docker stack (deploy/ls/ps/services etc)
 - K8S Operation
 - Service
 - Replication Controller
 - Deployment + RS



“Swarm & K8S” We are one big community

K8S vs Docker what is the difference ?

- Auto Scale (K8S Horizontal Pod Autoscaler (HPA))



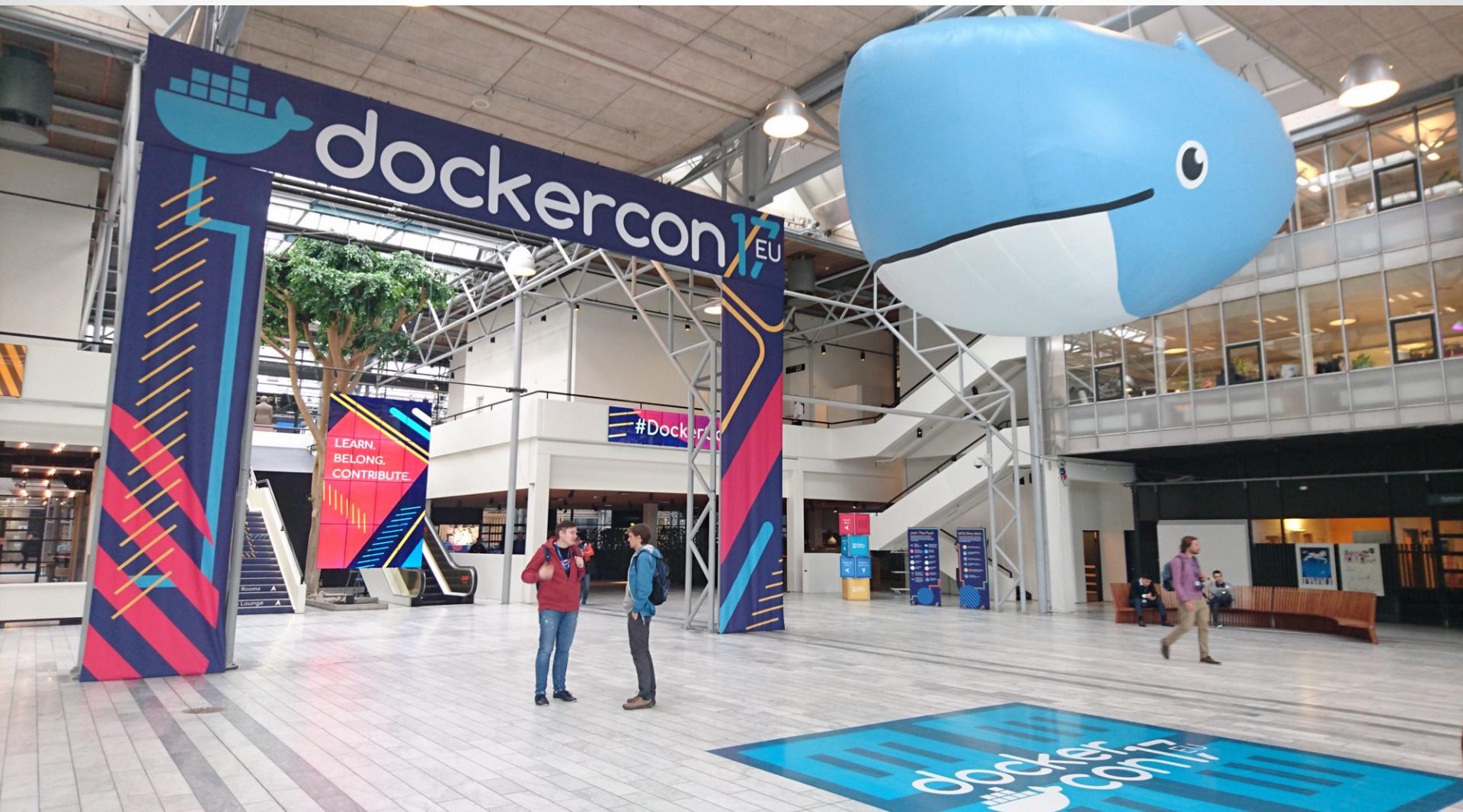
K8S vs Docker what is the difference ?

- Health Check
 - Docker: monitor by service on container
 - Kubernetes: monitor by service + customize
 - Liveness: Pods/Container is still alive (if not restart pods)
 - Readiness Pods/Container is ready to use (if not redirect workload to another pods)
 - Technic for check (Both Liveness/Readiness):
 - ExecAction:
 - TCPSocketAction:
 - HTTPGetAction:

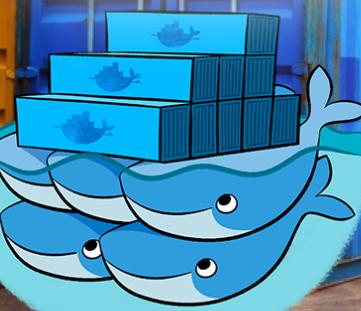


“Swarm & K8S” We are one big community

Demo Session



"Swarm & K8S" We are one big community



By Praparn Luengphoonlap
Email: eva10409@gmail.com

"Swarm & K8S" We are one big community

Q&A