DYALOG

Belfast 2018

Workshop TP2
Docker-Compose
Kubernetes

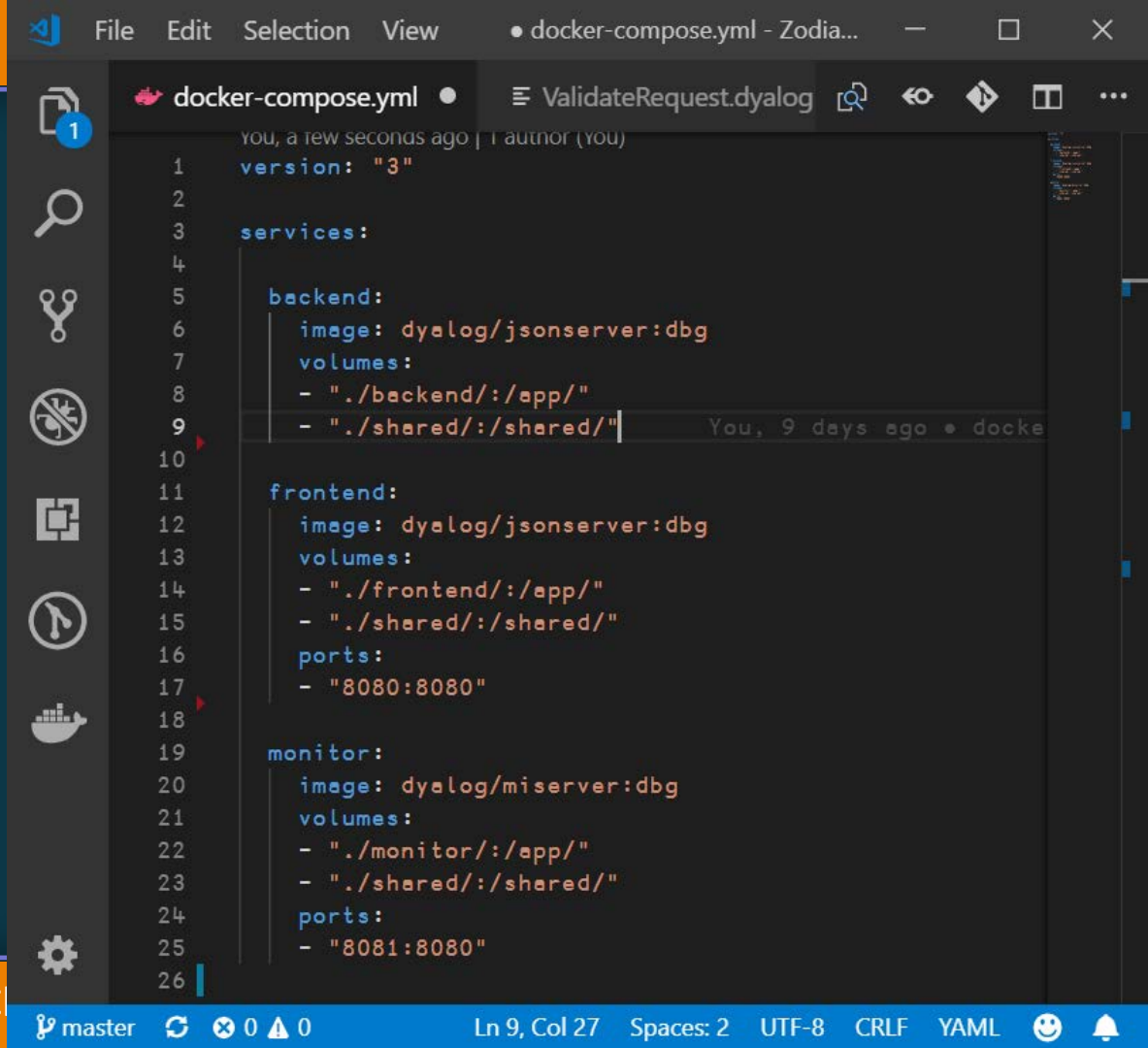
Morten Kromberg, CXO, Dyalog

Docker-Compose

- A docker-compose.yml file allows you to define a group of services that work together
- docker-compose will run the containers within a "virtual network"
- Local DNS allows each service to refer to the others by name

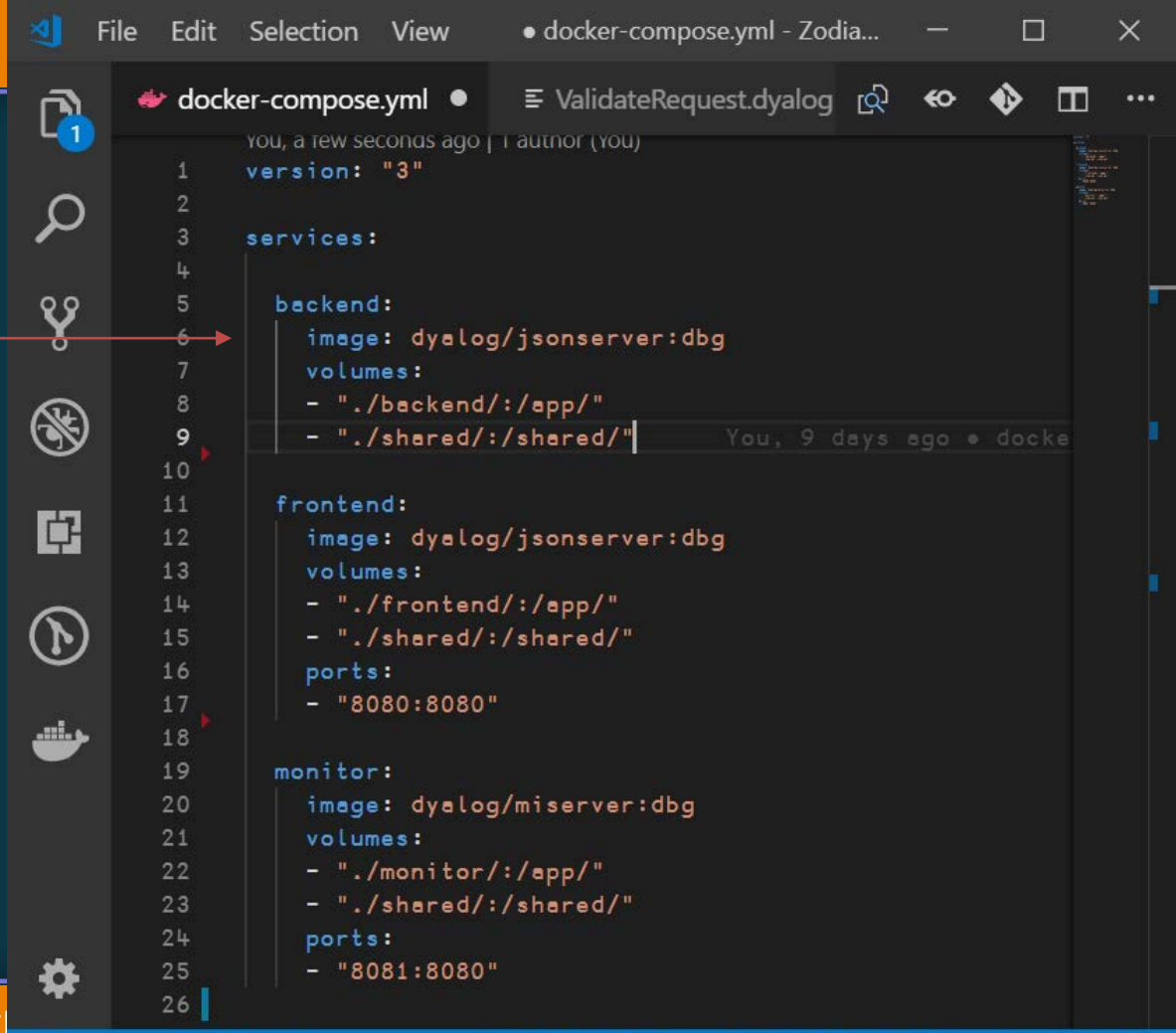
DYALOG

docker-compose



docker-compose

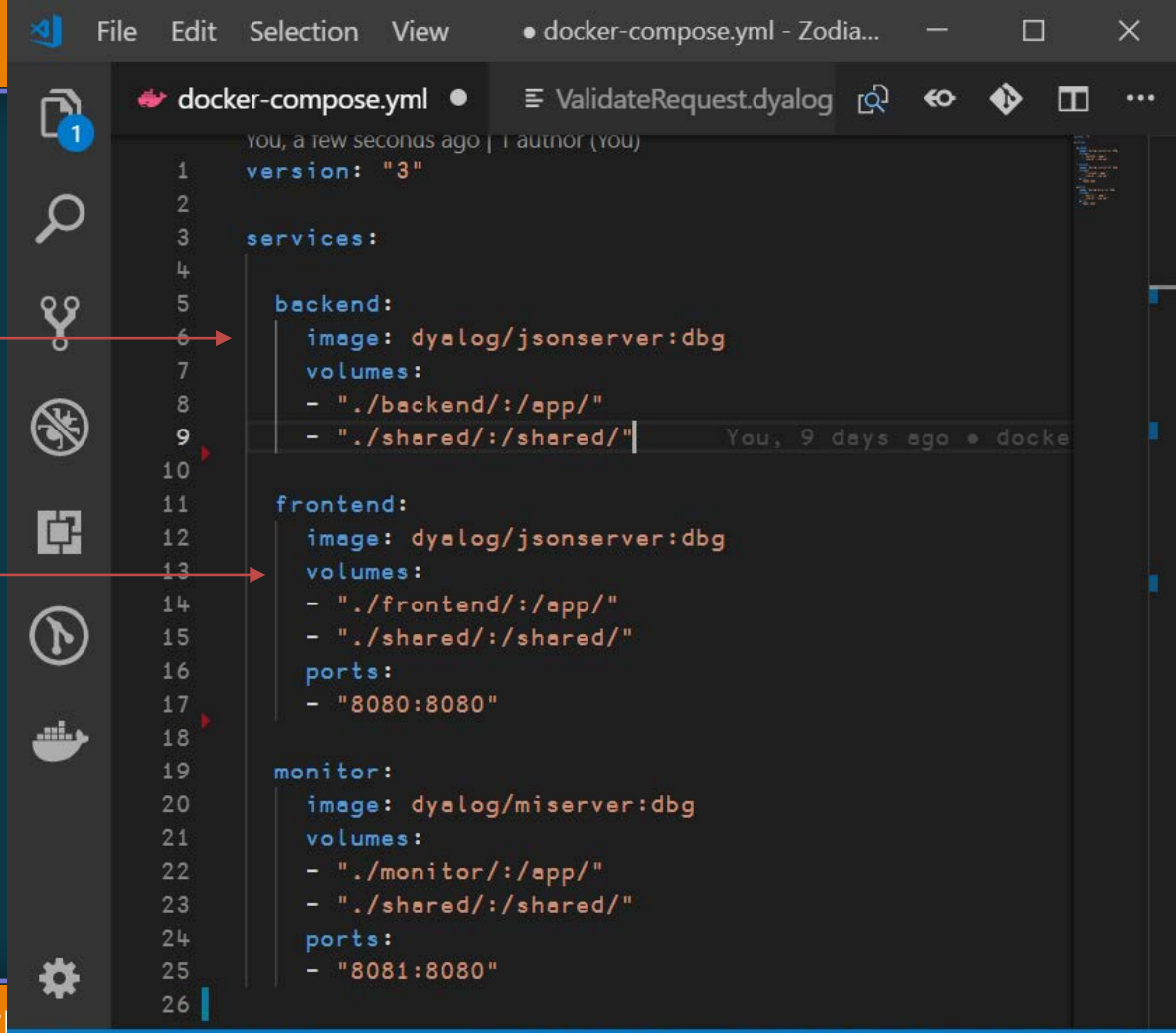
Image



docker-compose

Image

-v (volume mount)

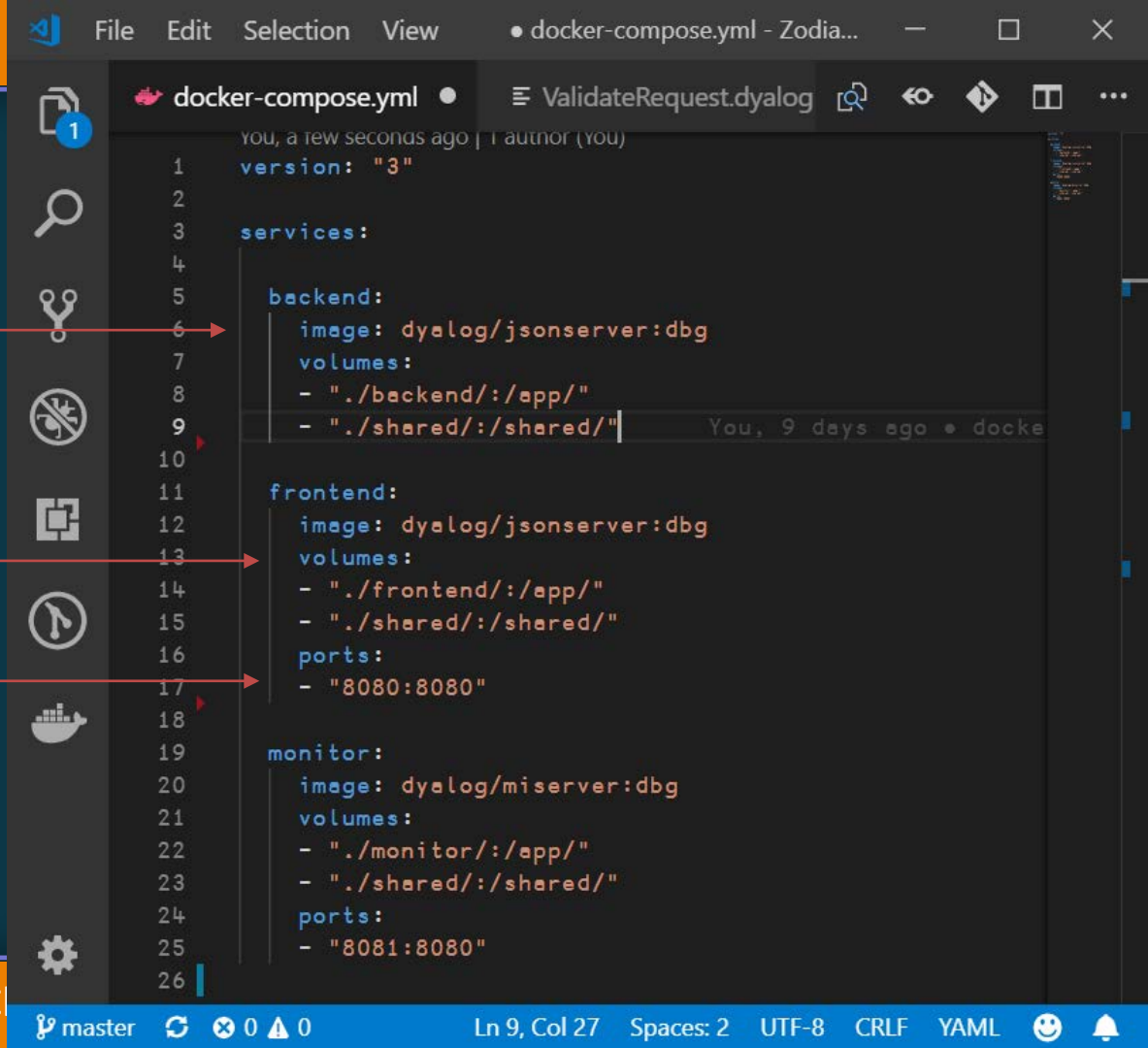


docker-compose

Image

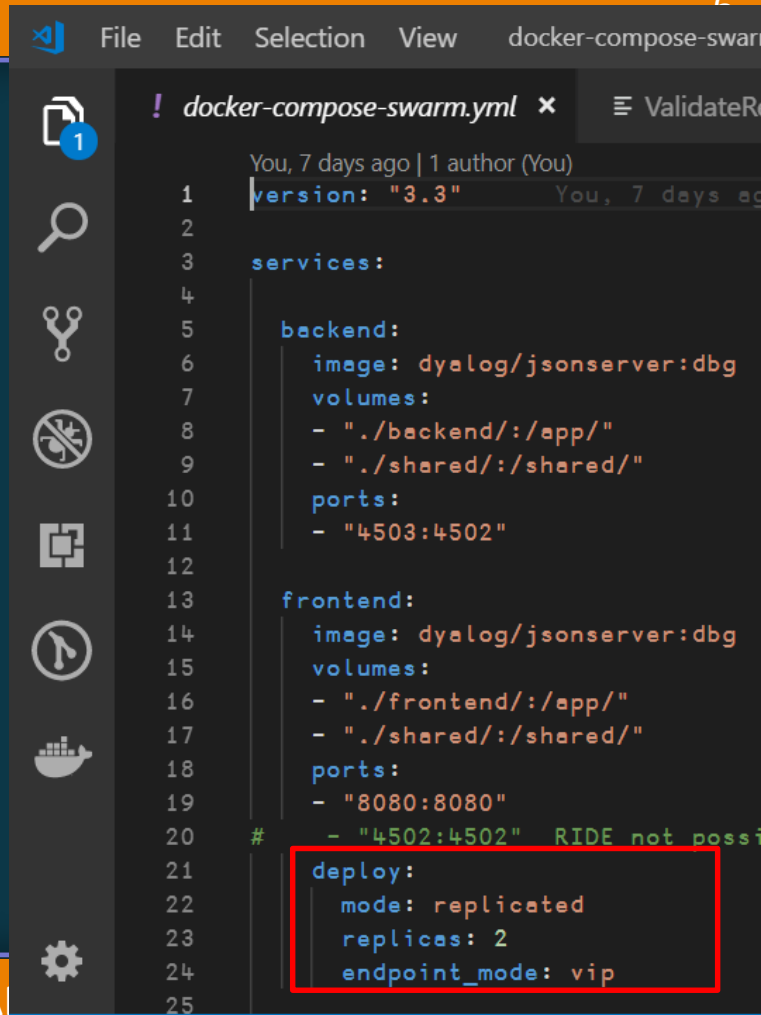
-v (volume mount)

-p (port maps)



docker-compose

- Docker-compose provides simple replication features
- Each replica thinks it is the only listener on port 8080
- Load balancers using virtual networking to switch between
- A selection of load balancers is available, the default is "round robin".



Docker Swarm

docker-compose services can be deployed under "Docker Swarm" for additional scaling support:

```
docker swarm init
docker node ls
docker stack deploy --compose-file docker-compose-swarm.yml zodiacservice
docker stack rm zodiacservice
docker service scale zodiacservice_frontend=3
docker swarm leave –-force
```

Docker-compose, Swarm and Kubernetes are evolving very rapidly and I ran out of time researching them.

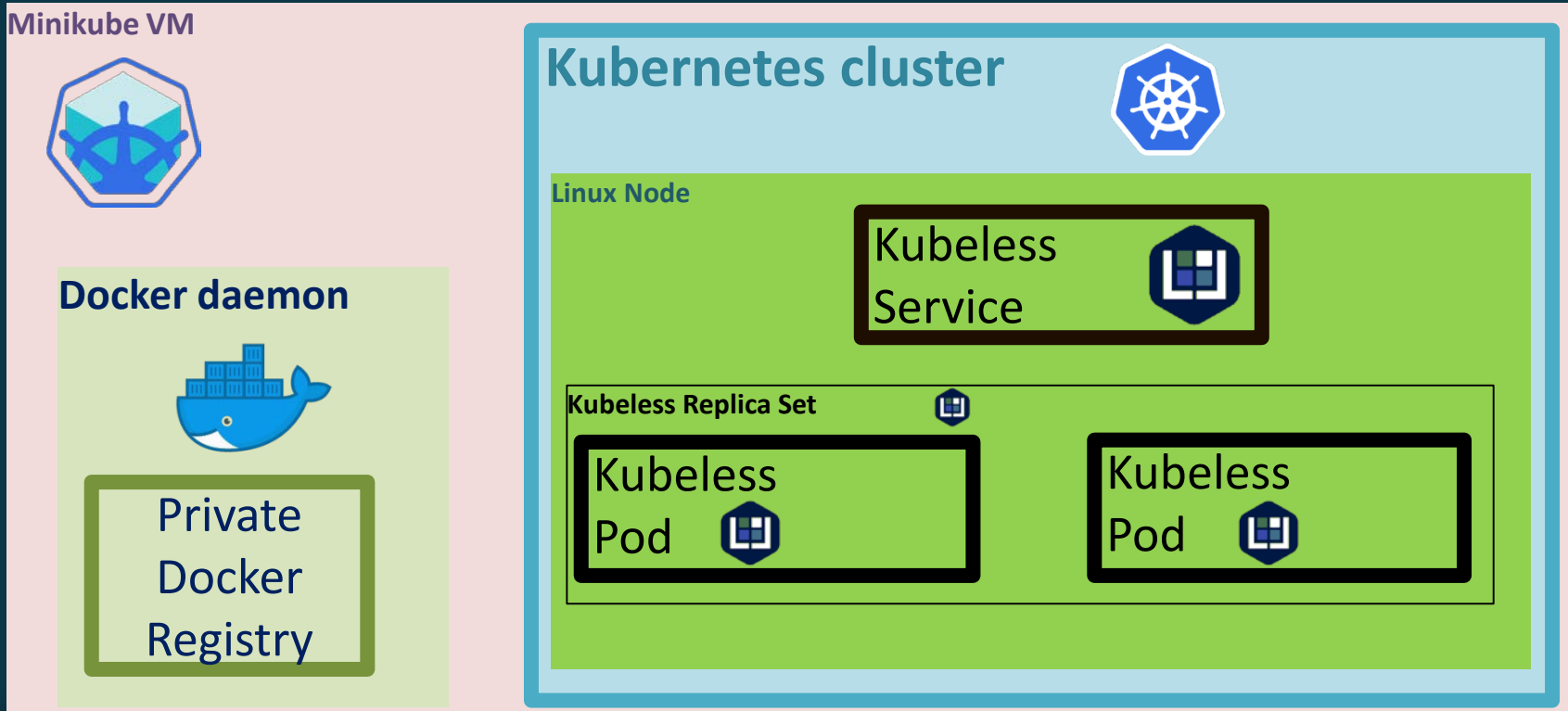
Exercise

- Experiment with docker-compose and docker swarm.
- Run 4 copies of the frontend
- Verify using the monitor

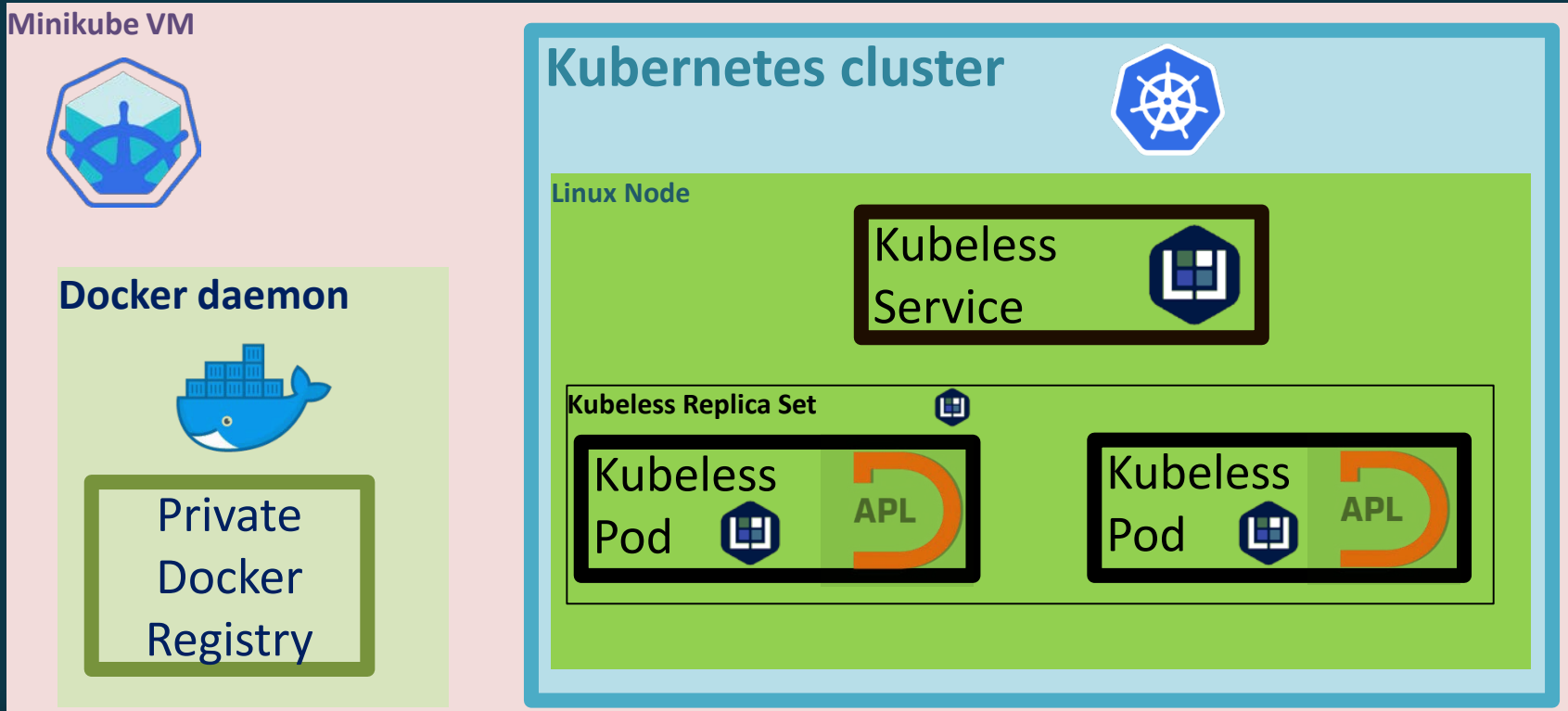
Exercise / Project

- Complete the ZodiacService application to
  - record calls in a shared component file
  - enhance the monitor to report usage of each frontend instance in a replicated deployment

Marko's "Serverless Laboratory"



Marko's "Serverless Laboratory"



Kubernetes

- The equivalent of a docker-compose file is called a "Pod"
- Kubernetes manages a number of nodes (1 master + 0 or more)
- Services can be replicated within a Pod, and Pods can be replicated across the network of nodes.
- This turned out to be too difficult to set up for/in a 3 hour workshop ☺