
grape up
KubeCon
CloudNativeCon
North America 2019


KubeCon
CloudNativeCon
North America 2019
Kubernetes in Your 4x4
Continuous Deployment Directly to the Car
Rafał Kowalski

Rafał Kowalski

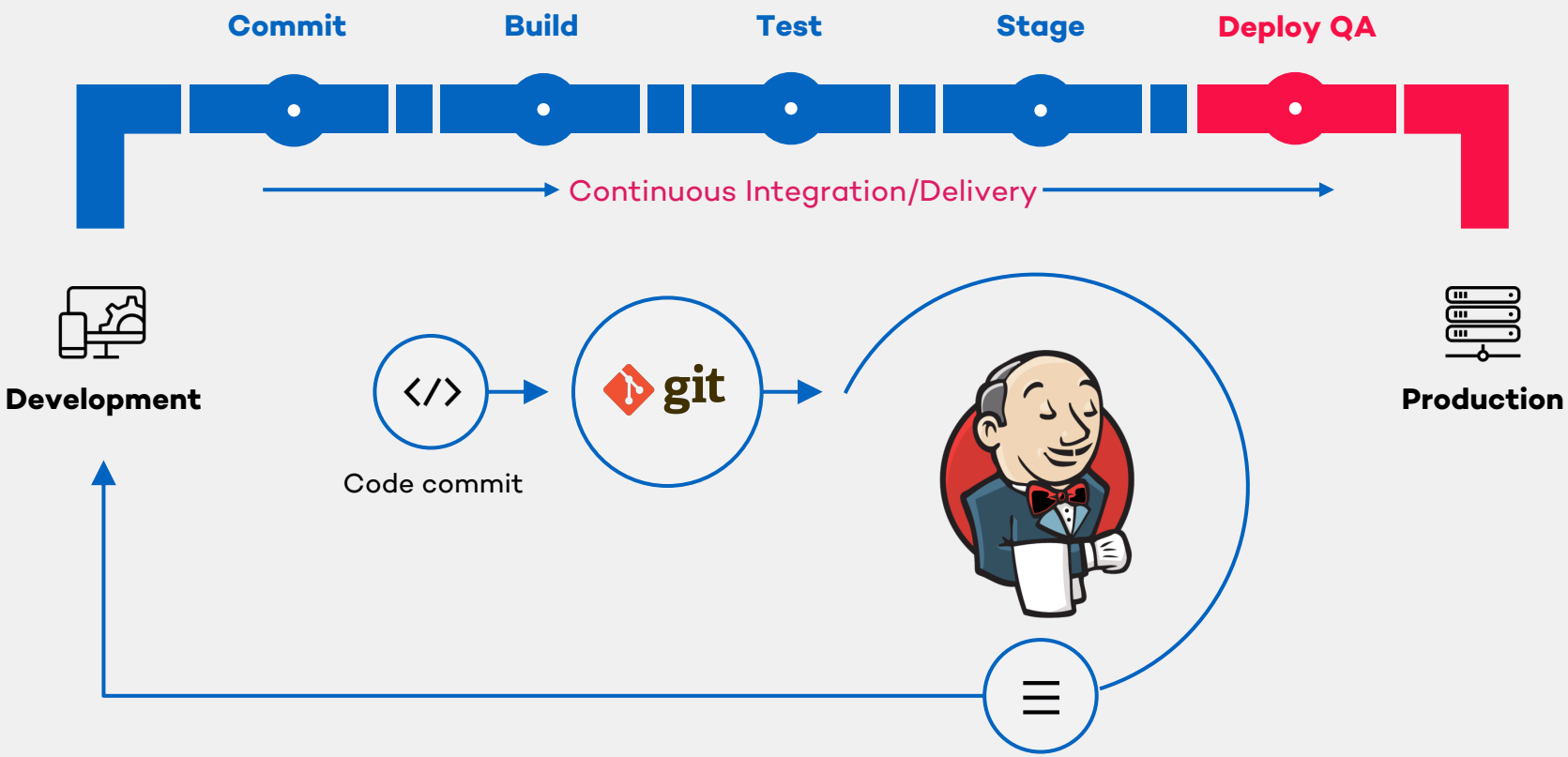
Cloud Solution Architect @ Grape Up
PhD @ Polish Academy of Sciences

@b3rnoulli

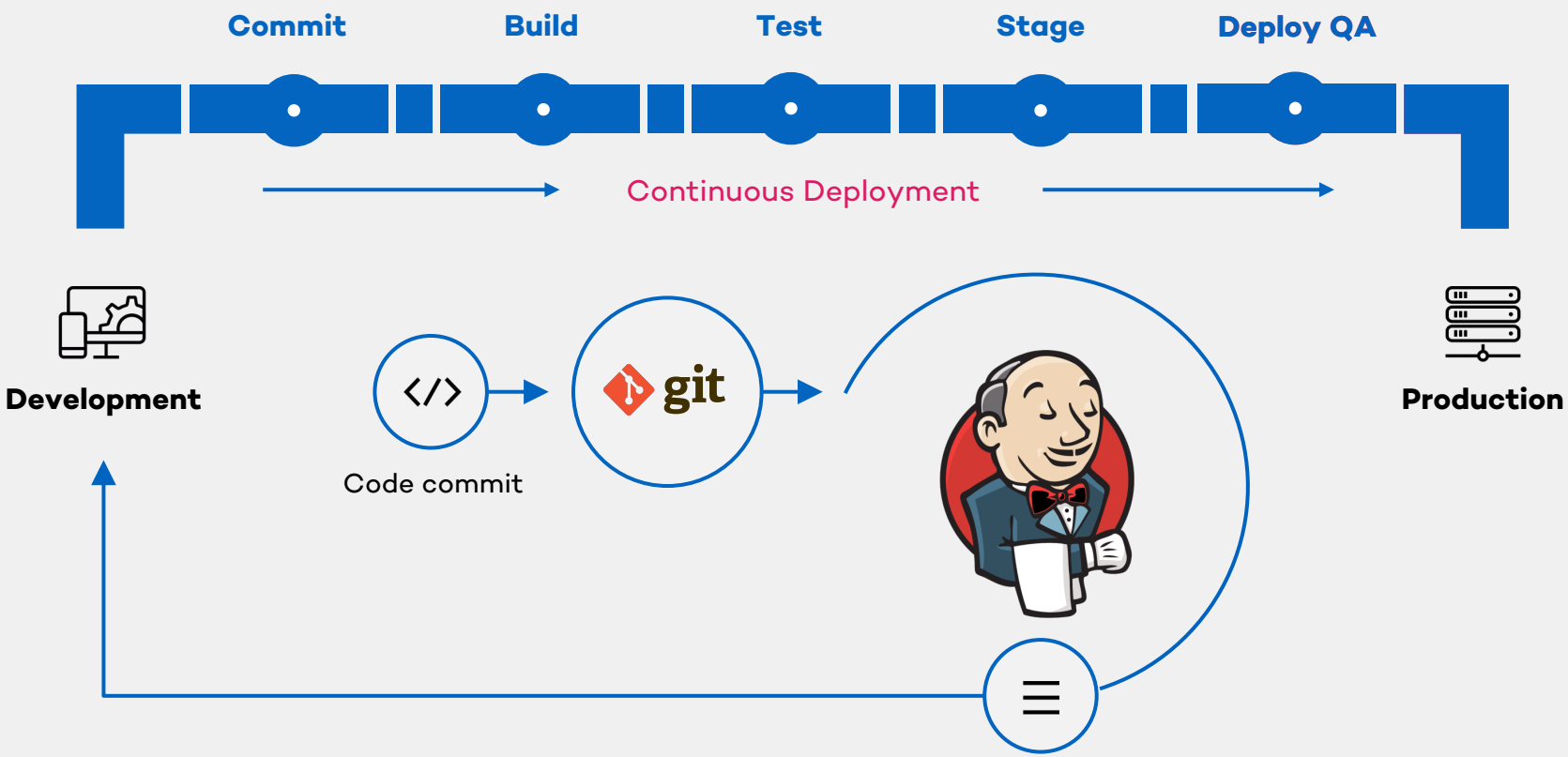
b3rnoulli

Continuous Delivery

grape up

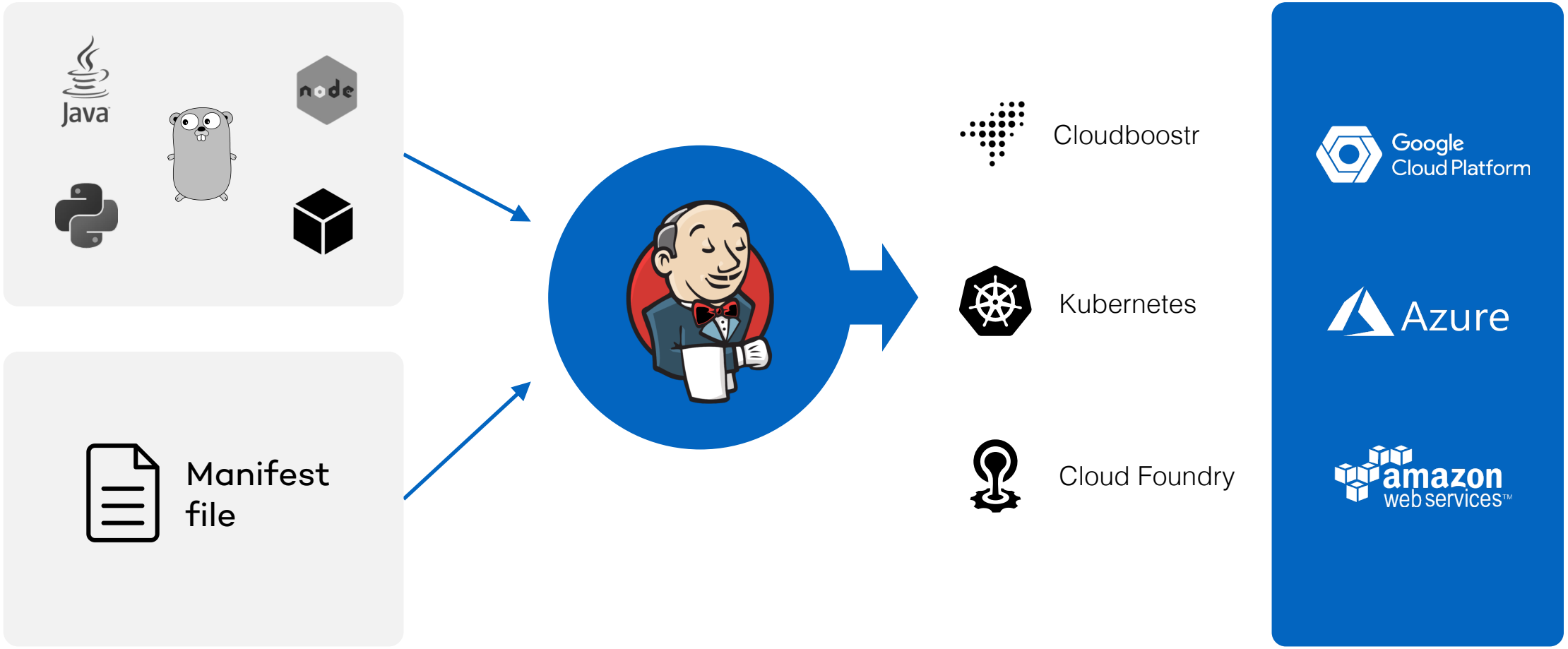


Continuous Deployment



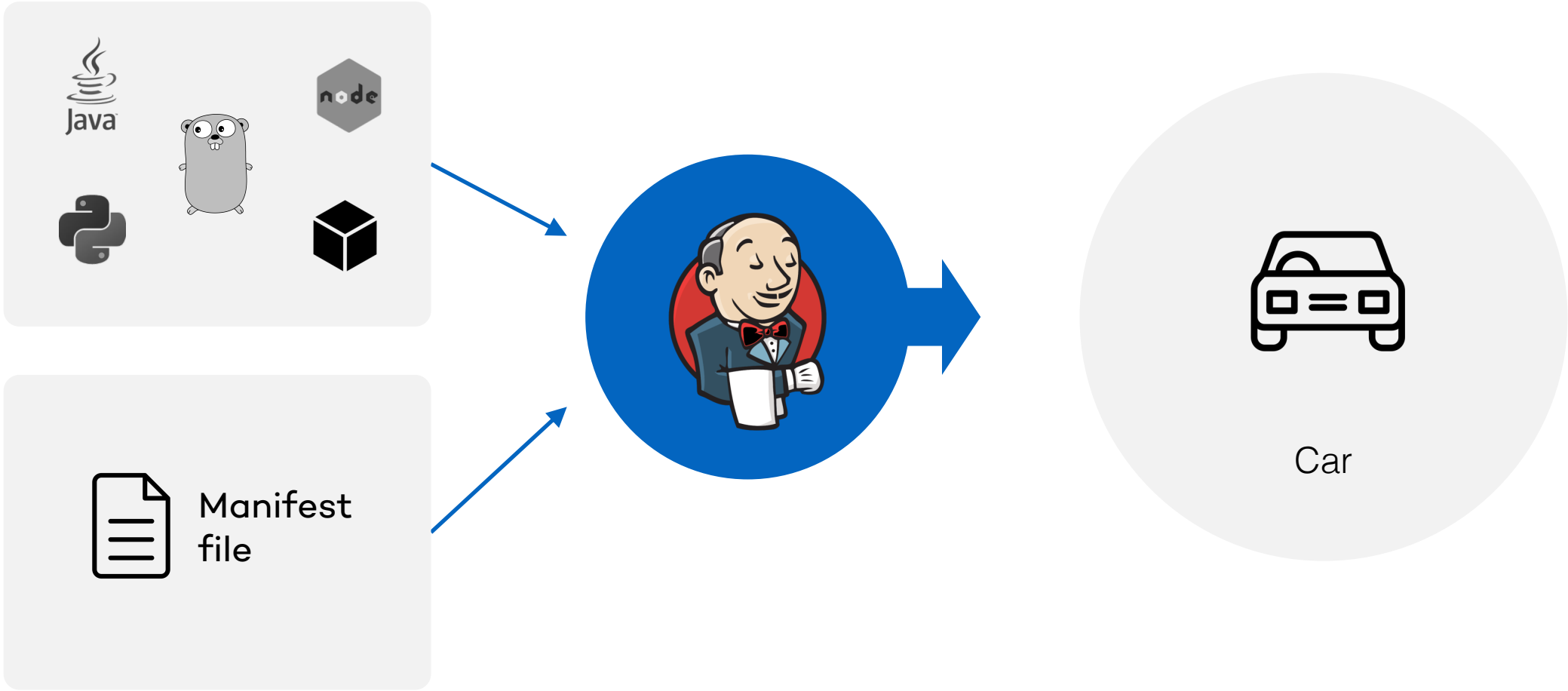
Deployment in the Cloud

grape up



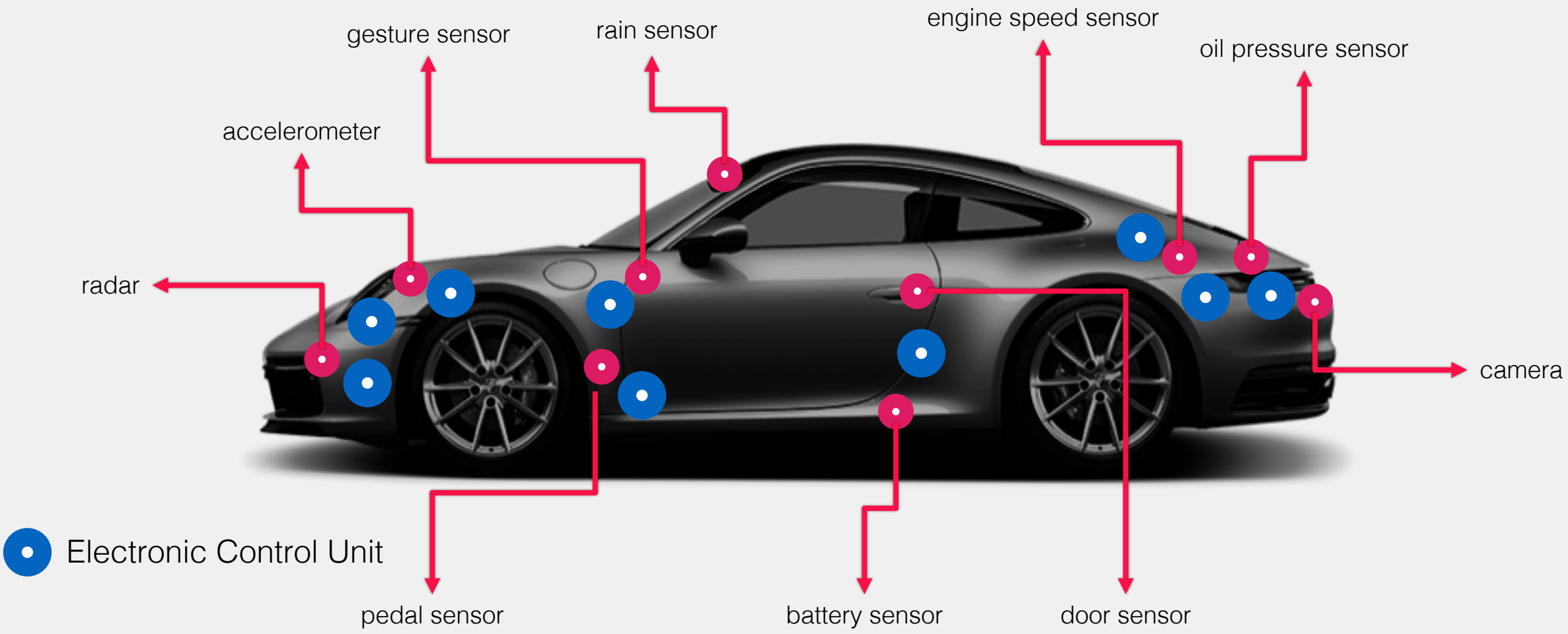
Deployment to the Car

grape up



Car architecture

grape up



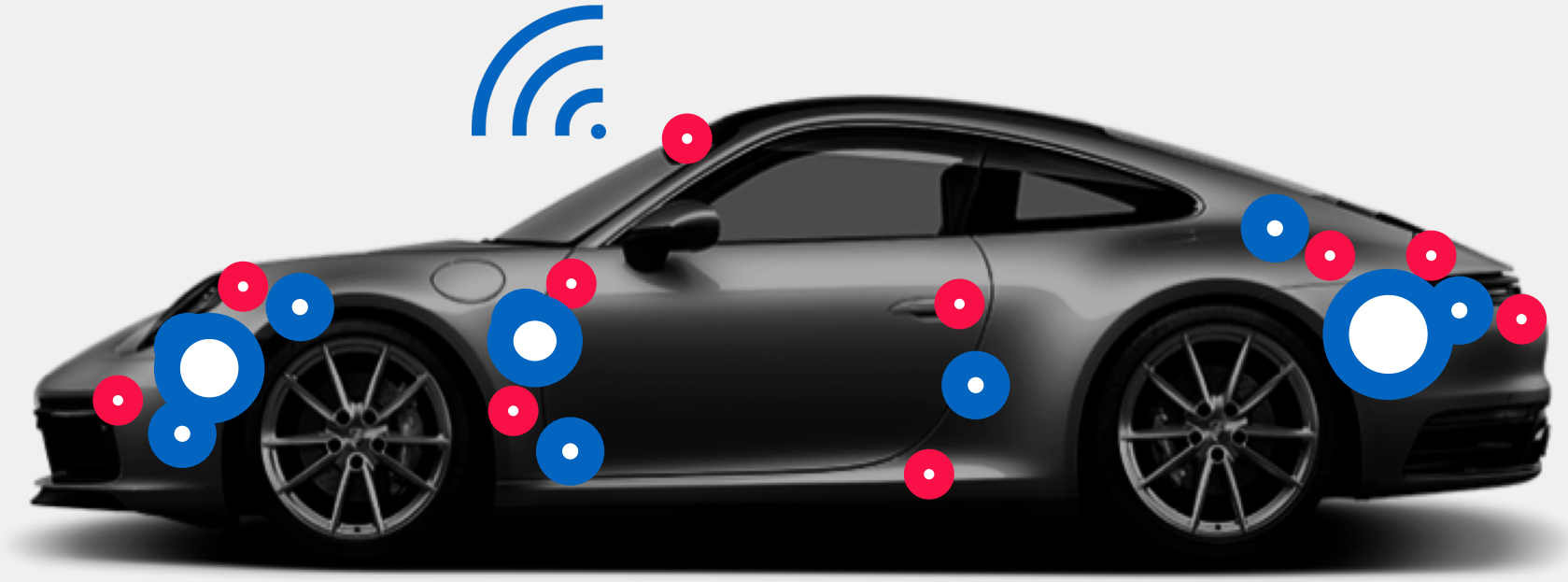
Old fashined deployment

grape up

0 1 0 1 1 1 0 1 0 0 .....

Consolidation of Electronic Control Units in modern cars

grape up



High Performance ECU

Connected Car

grape up

0 1 0 1 1 1 0 1 0 0 .....

Kubernetes in the car

Quite challenging

grape up

Most of Kubernetes distributions don’t support ARM

Kubernetes wasn’t designed for embedded software

Kubernetes consume a lot of resources (up to 4gb of RAM)

Kubernetes has a lot of features not required in edge devices

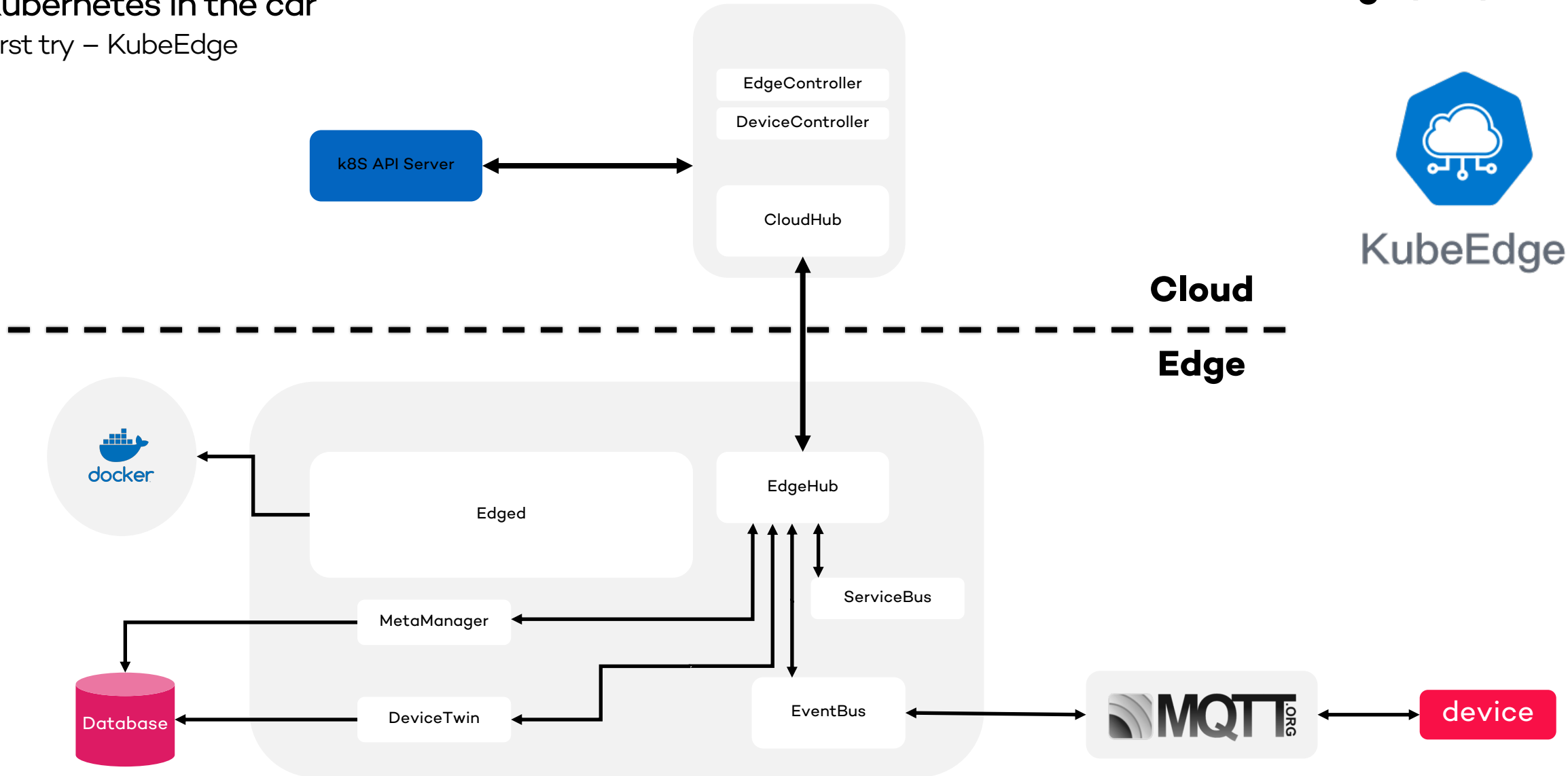
grape up

- Fits resource constraints environments
- Support offline operations
- MQTT-based communication
- SDK based Development for Device Addition etc.

Kubernetes in the car

First try – KubeEdge

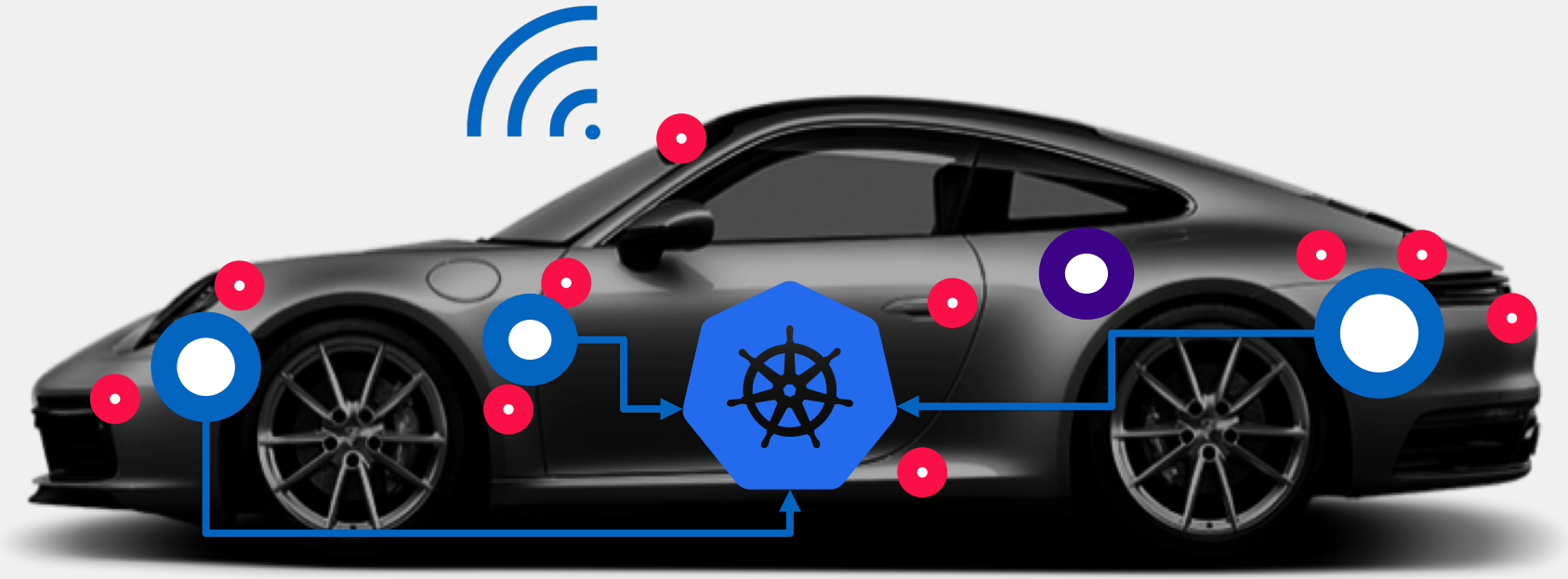
grape up



Kubernetes in the car

Revisited architecture

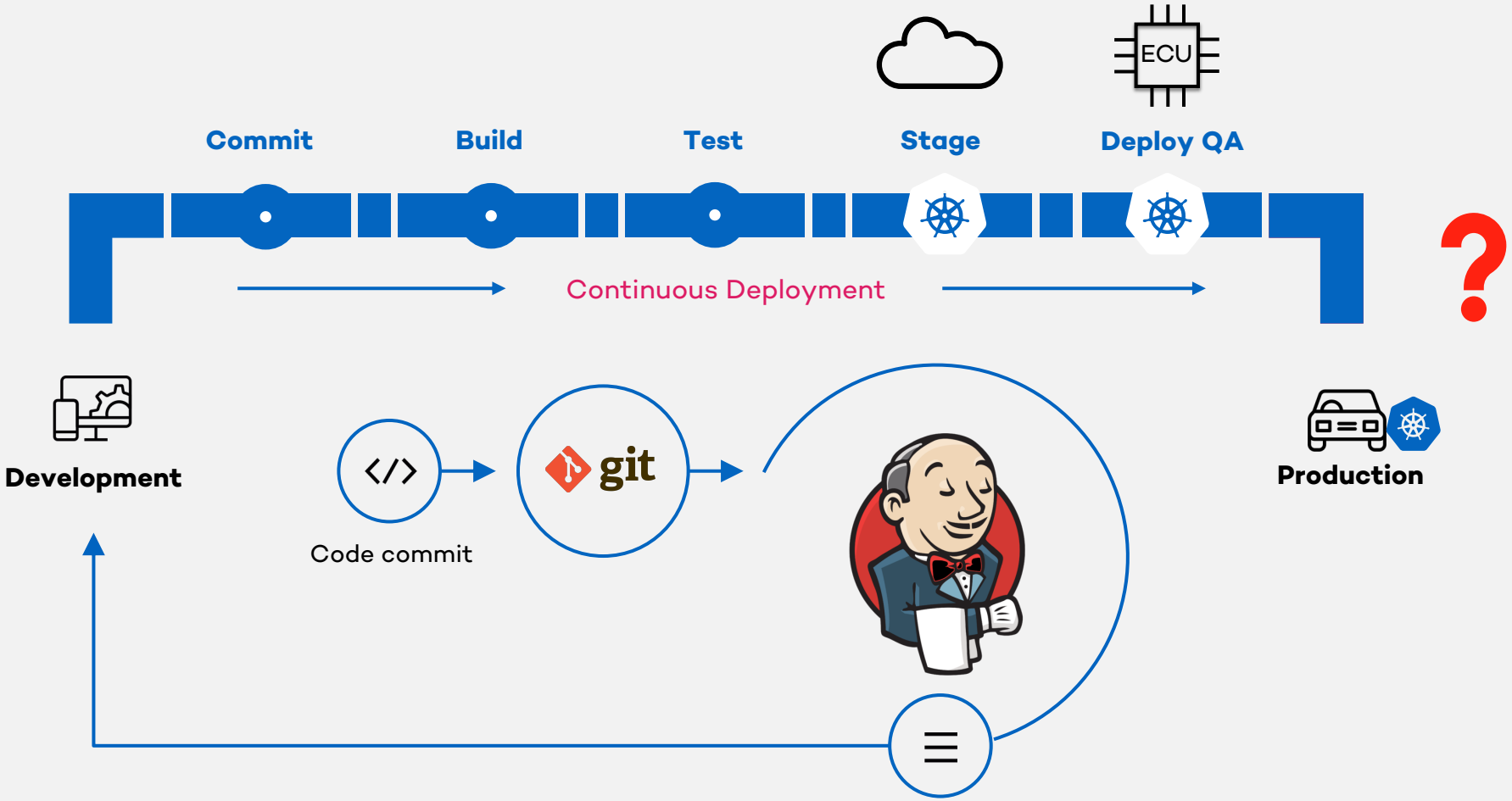
grape up



ECUs as workers

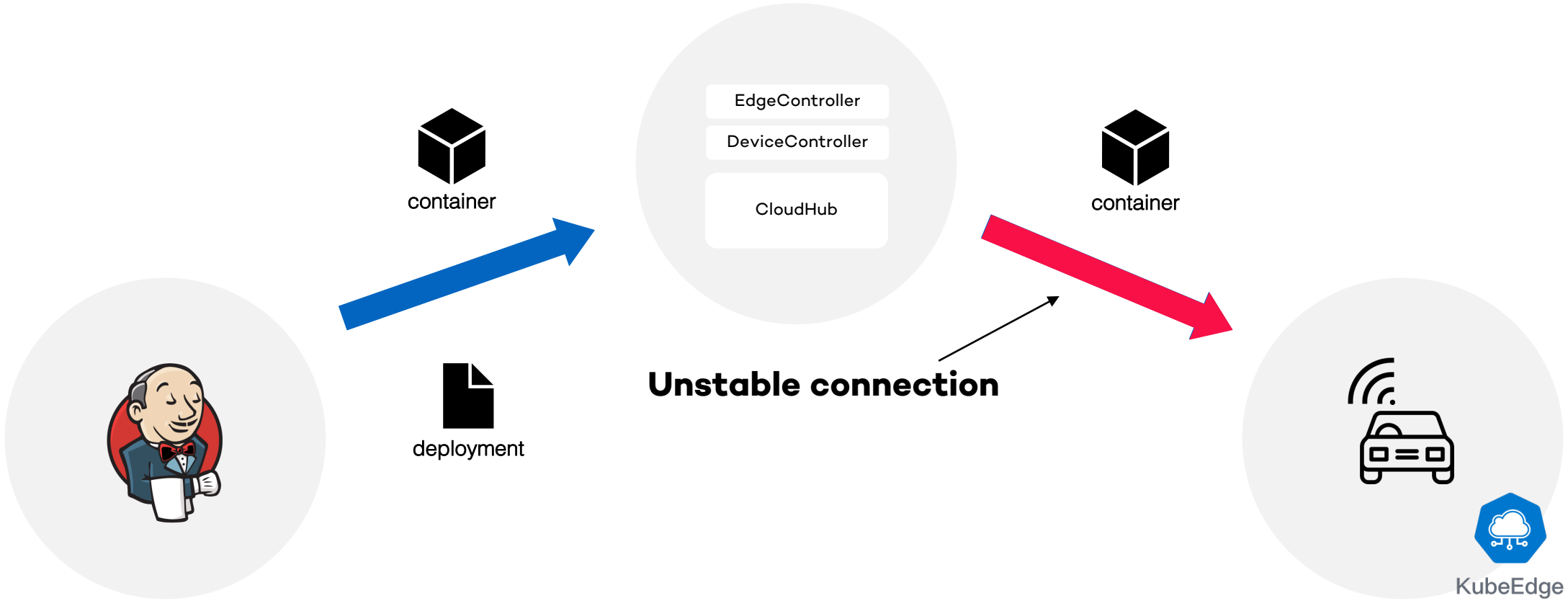
Let's build the pipeline

grape up



Let's build the pipeline

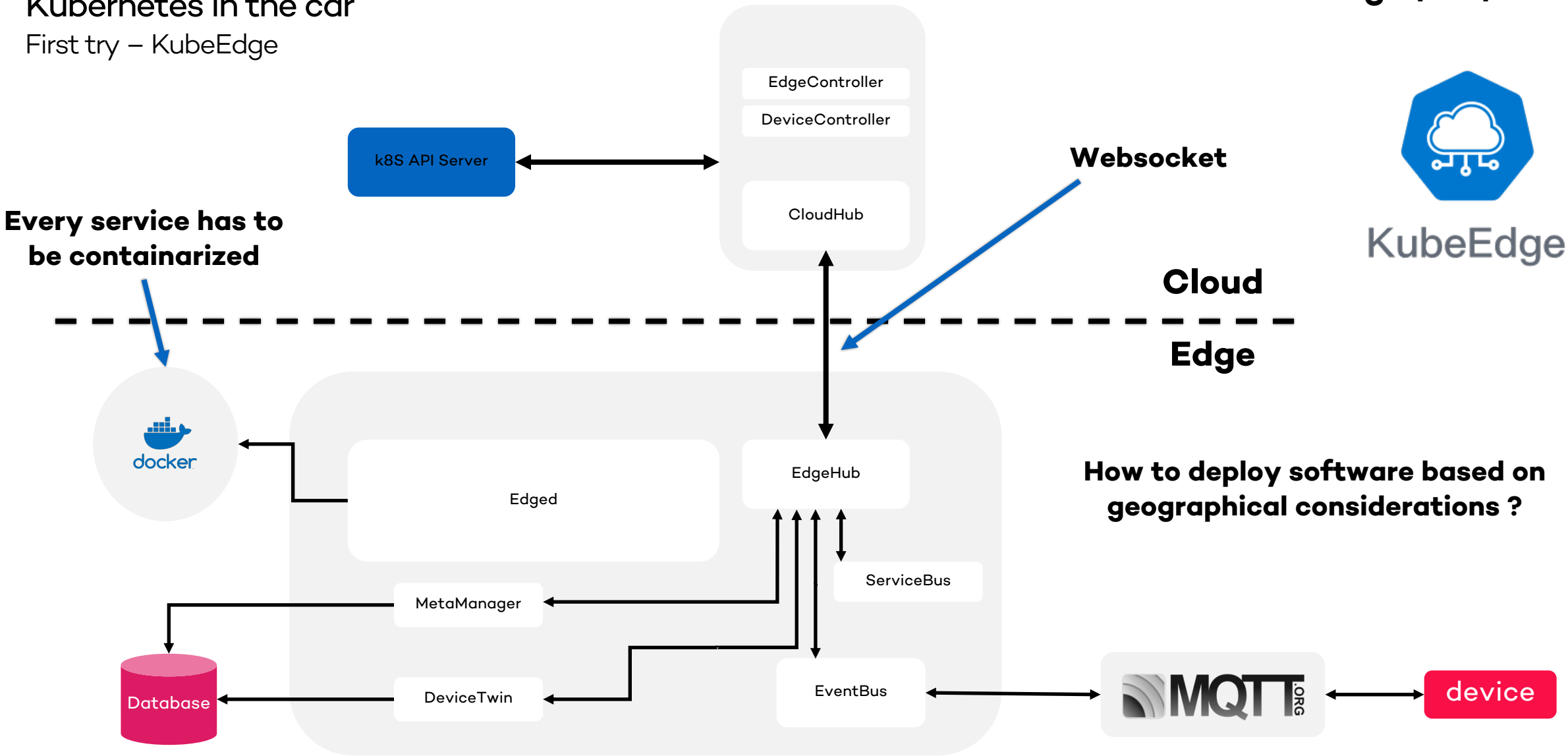
grape up



Kubernetes in the car

First try – KubeEdge

grape up



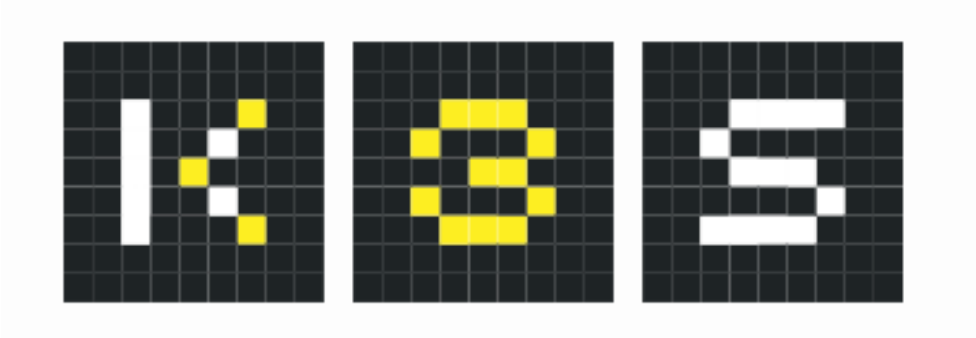
KubeEdge

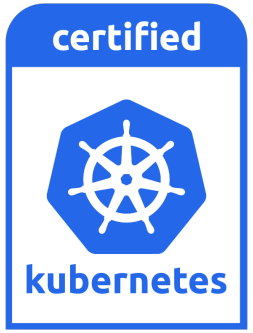
How to deploy software based on geographical considerations ?

Kubernetes in the car

Second try – k3s

grape up





- Kubernetes lightweight distribution
- Fits resource constraints environments
- Support offline operations
- 200MB disk space and 512MB RAM

Kubernetes in the car
Second try– k3s

grape up

Removes

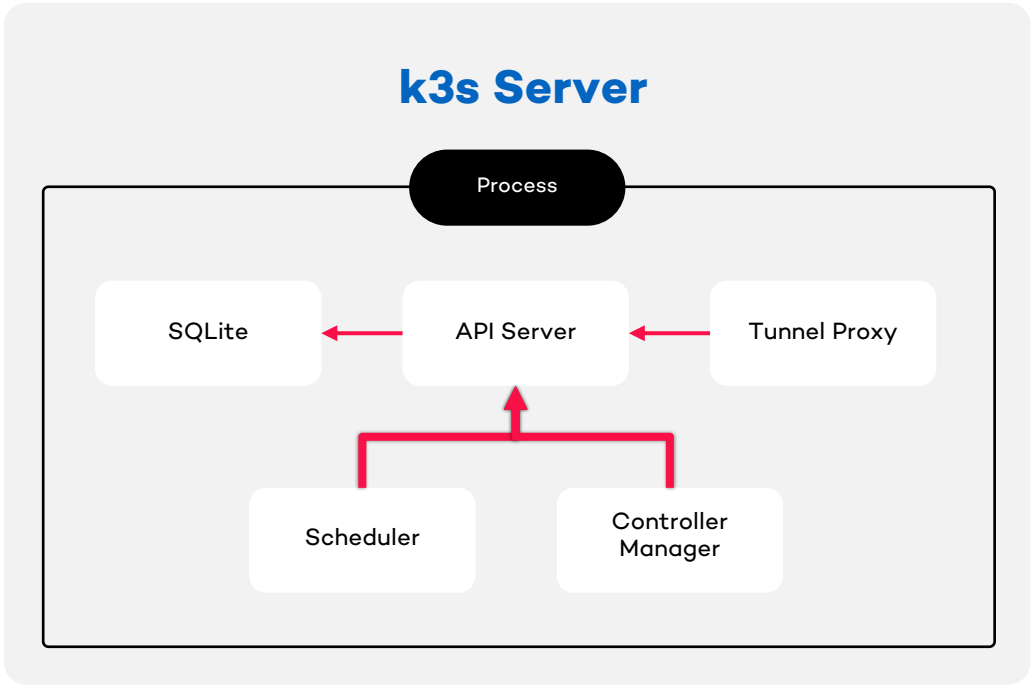
- Legacy and non-default features
- Alpha features
- In-tree cloud providers
- In-tree storage drivers
- Docker (optional)

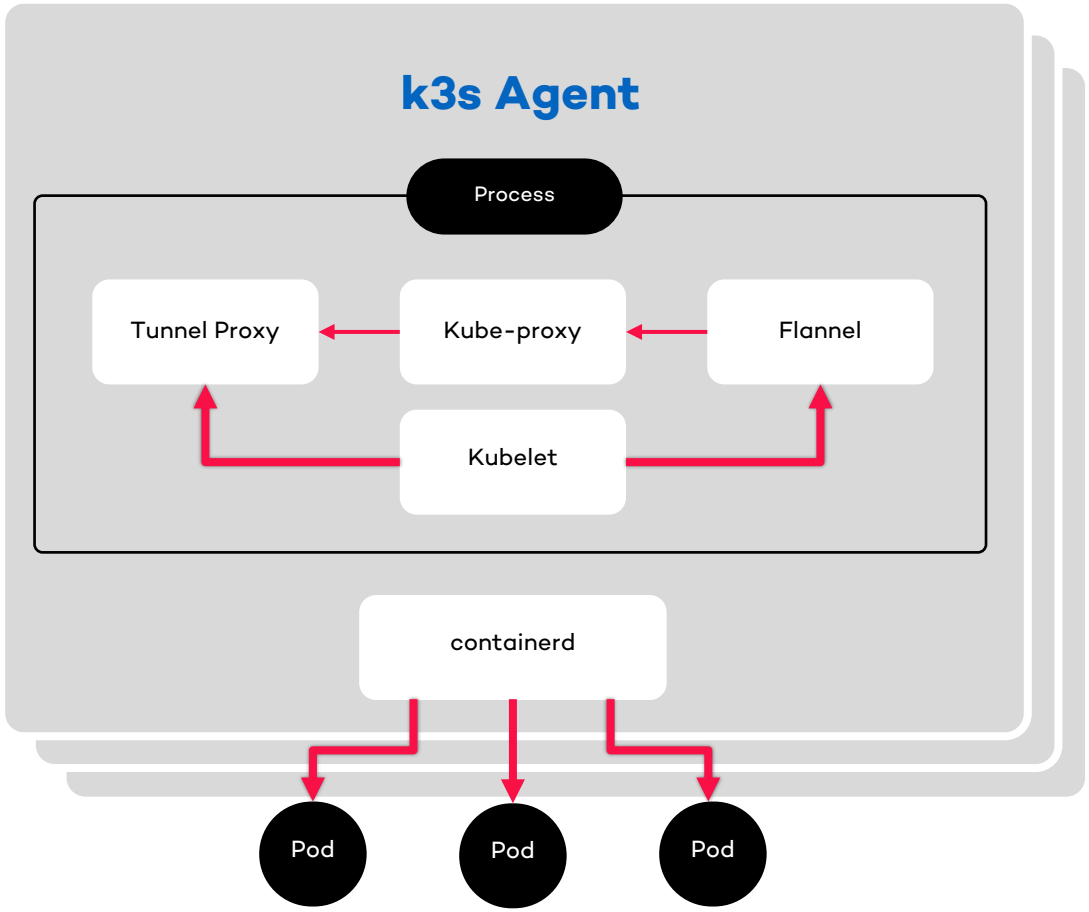
Adds

- Simplified installation
- SQLite3 support in addition to etcd
- TLS management
- Automatic Manifest and Helm Chart management
- containerd, CoreDNS, Flannel

k3s architecture

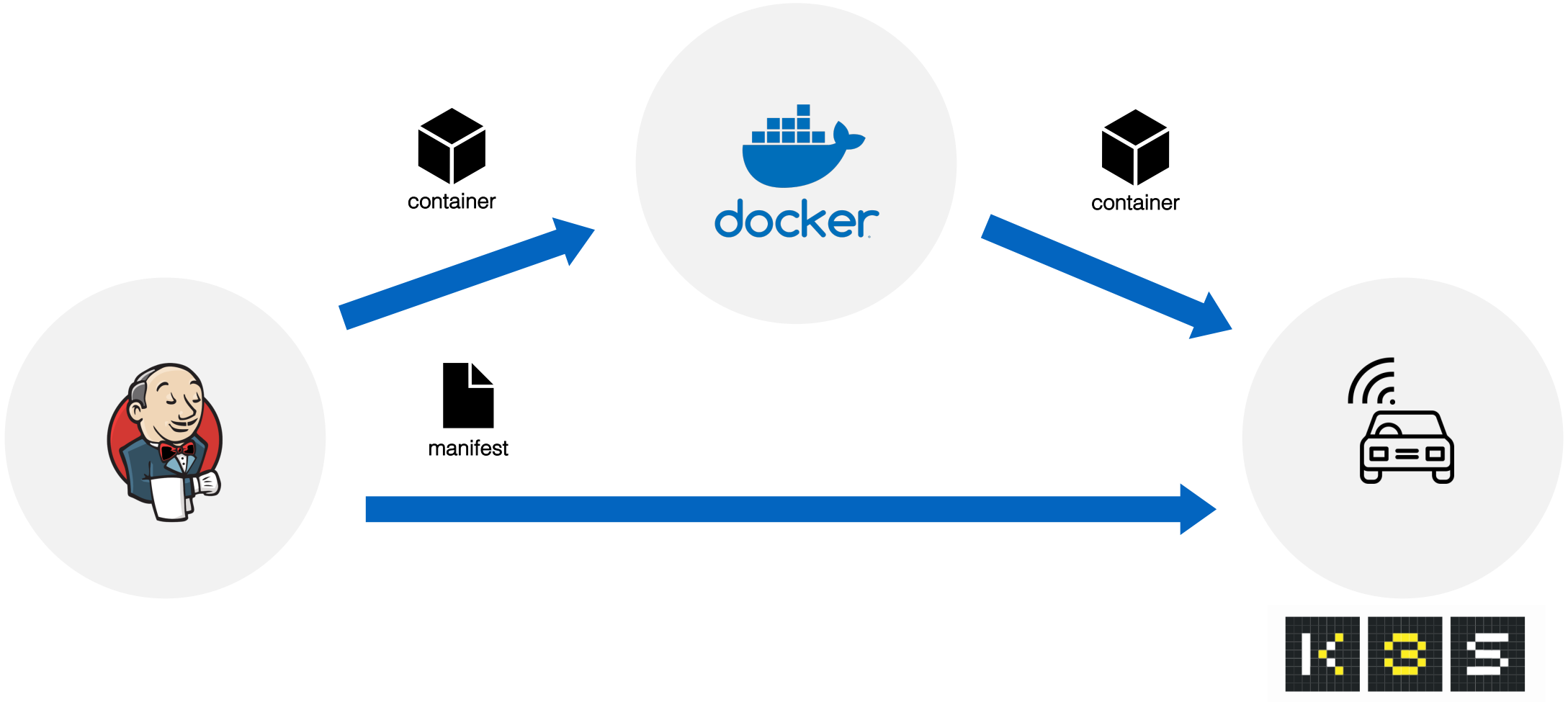
grape up





Let's build the pipeline

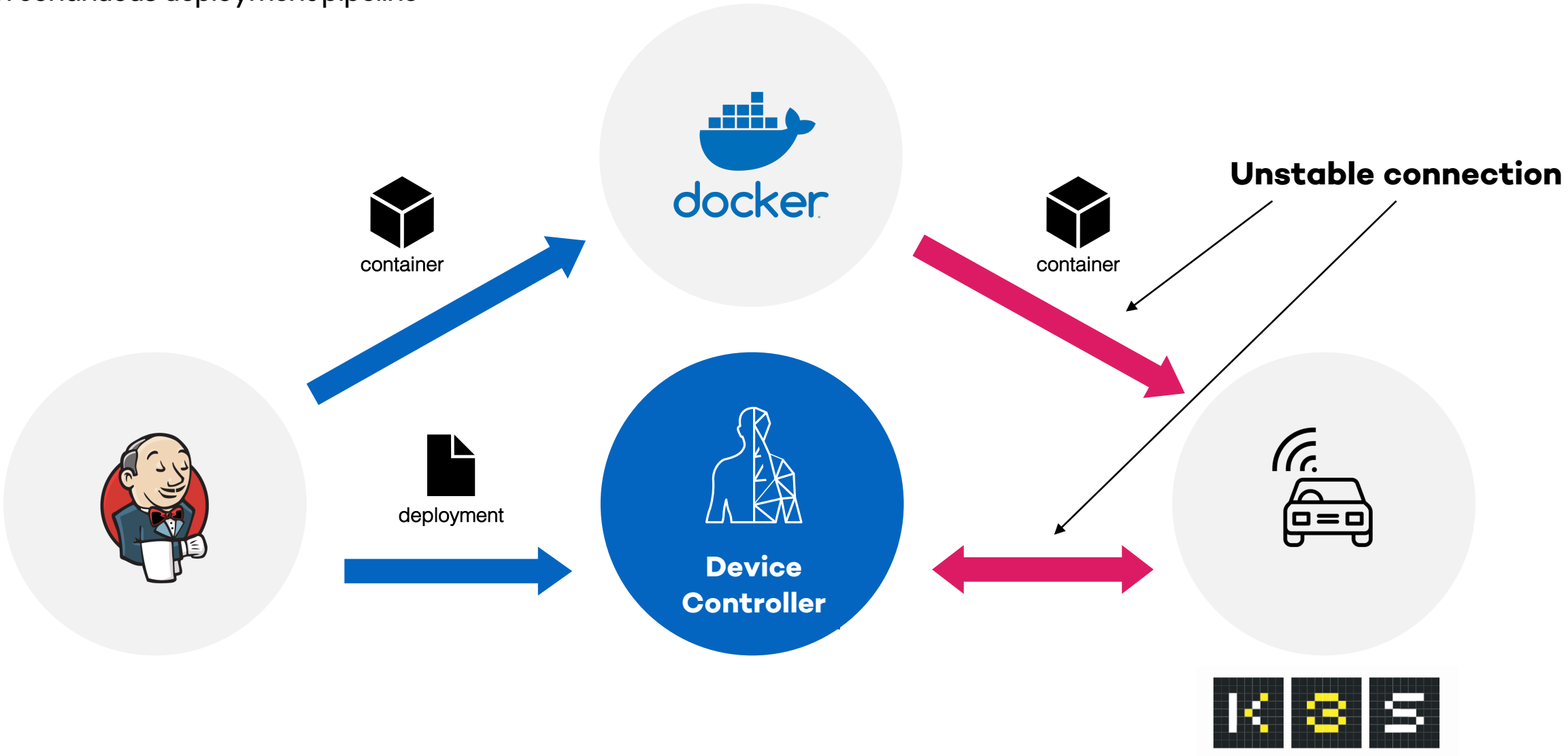
grape up



Digital Twin pattern

in continuous deployment pipeline

grape up

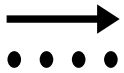


RSocket to the rescue!

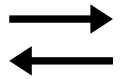
grape up

RSocket is **framed, message-based, binary,** bi-directional protocol , based on reactive streams back pressure and four-elements interaction model

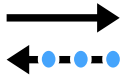
- Interaction is broken down into frames
- It can run on top of the TCP / Web Socket / Aeron
- **Payload could be anything – even large thing**
- Rich interaction model



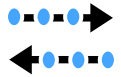
Fire & Forget



Request - Response



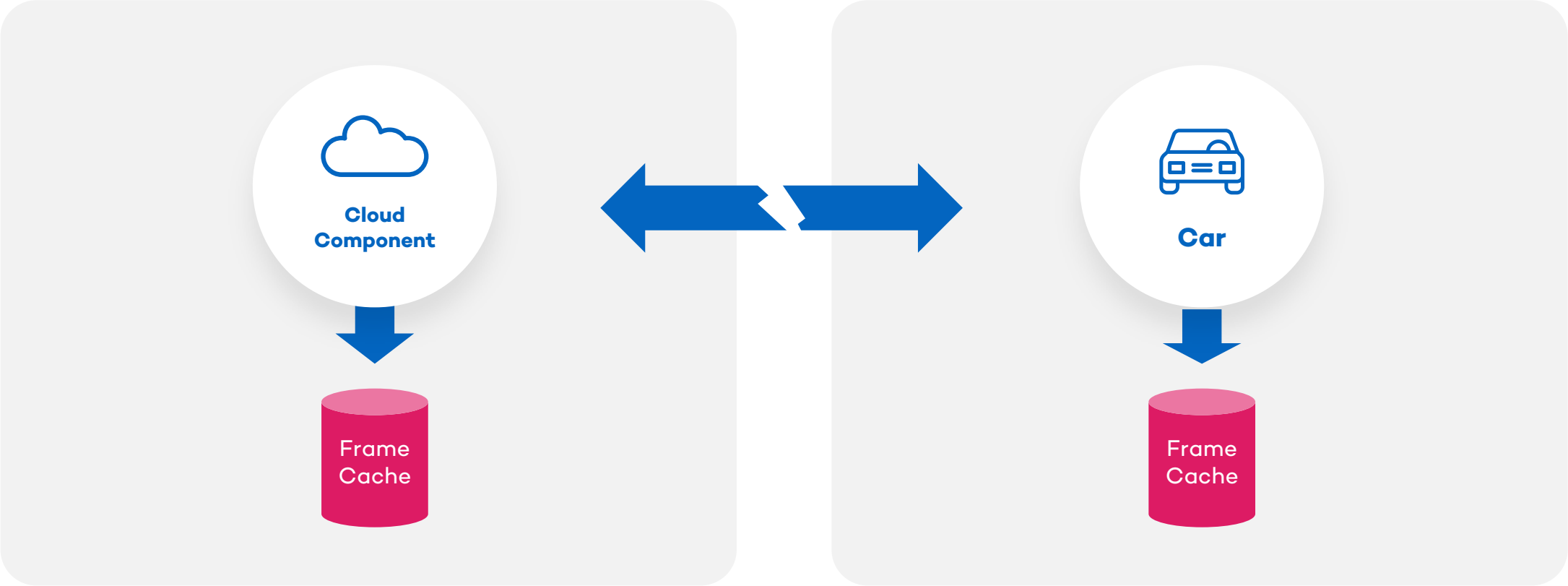
Request- Stream



Channel

Resumability in Socket

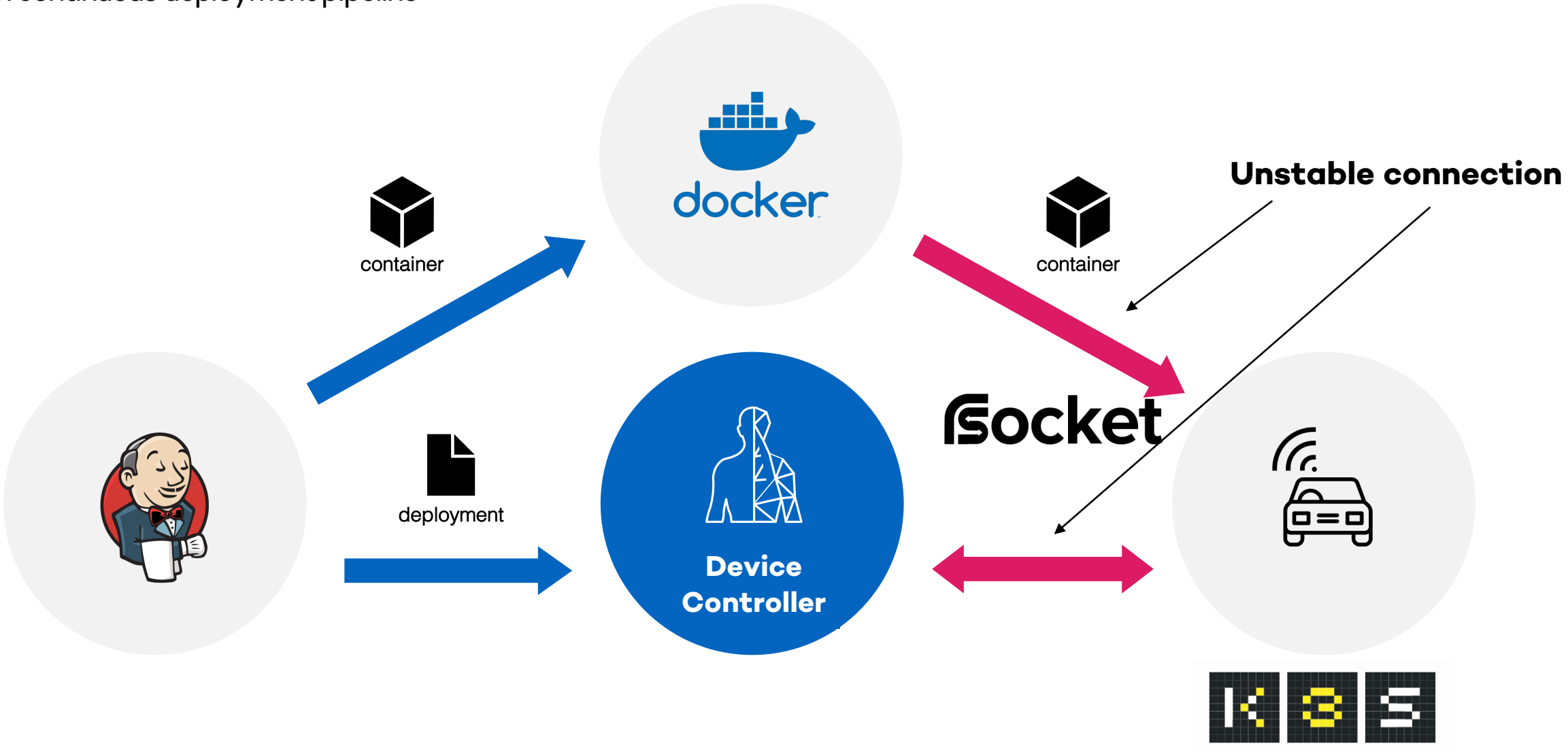
grape up



Digital Twin

in continuous deployment pipeline

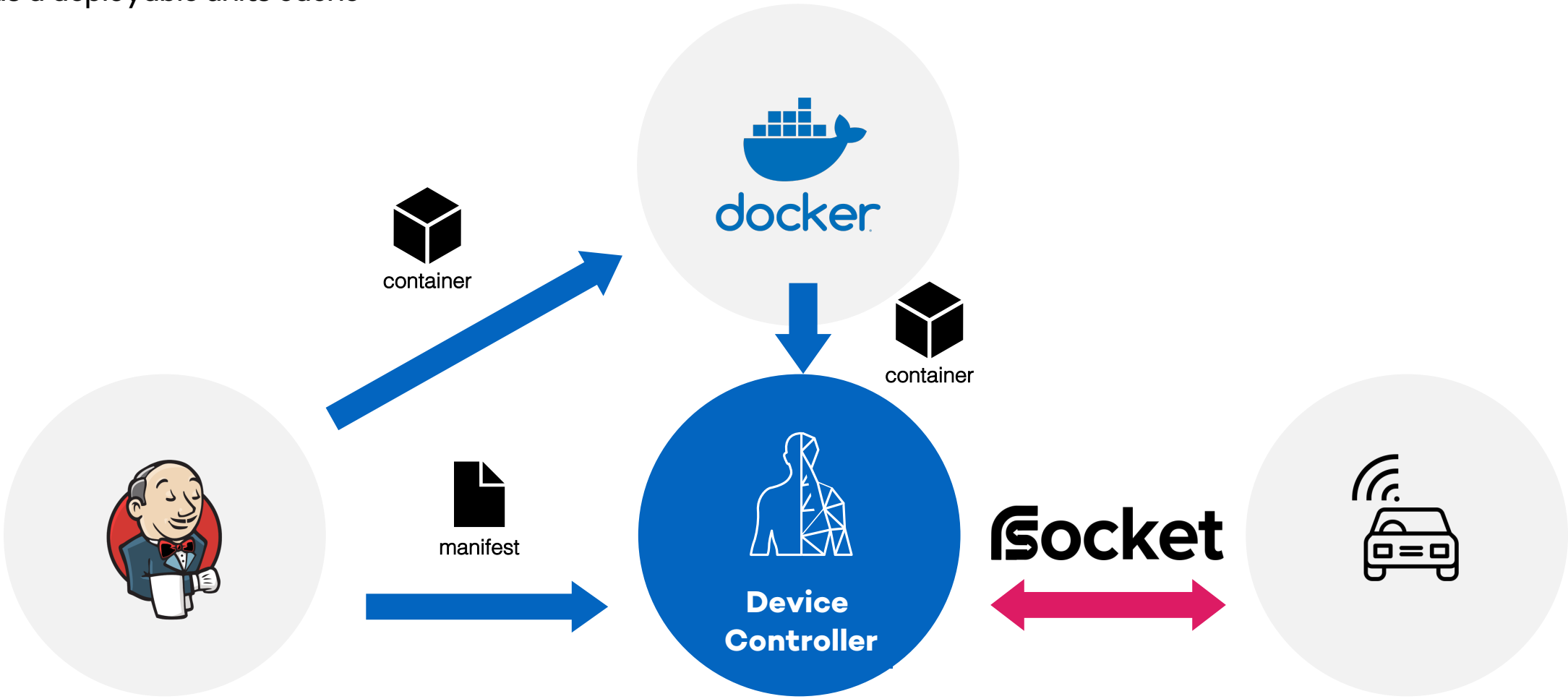
grape up



Digital Twin

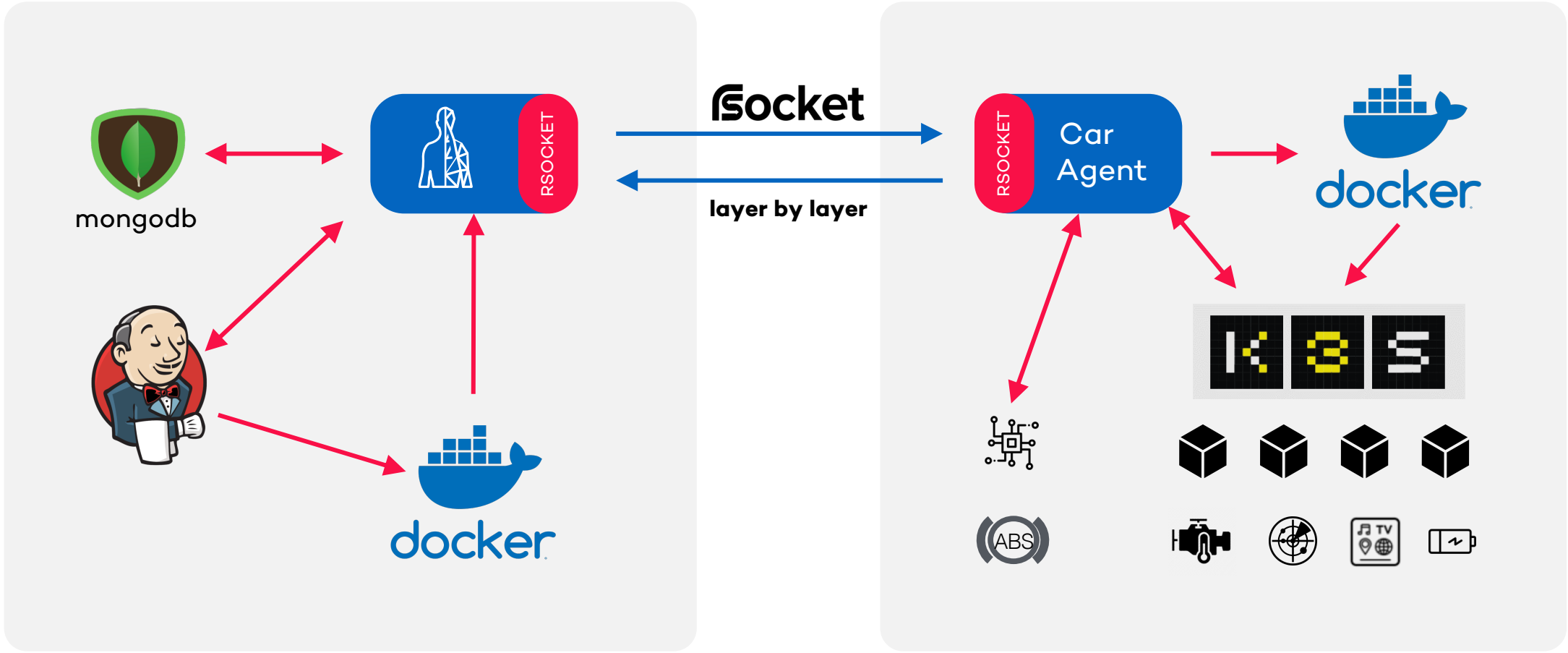
as a deployable units cache

grape up



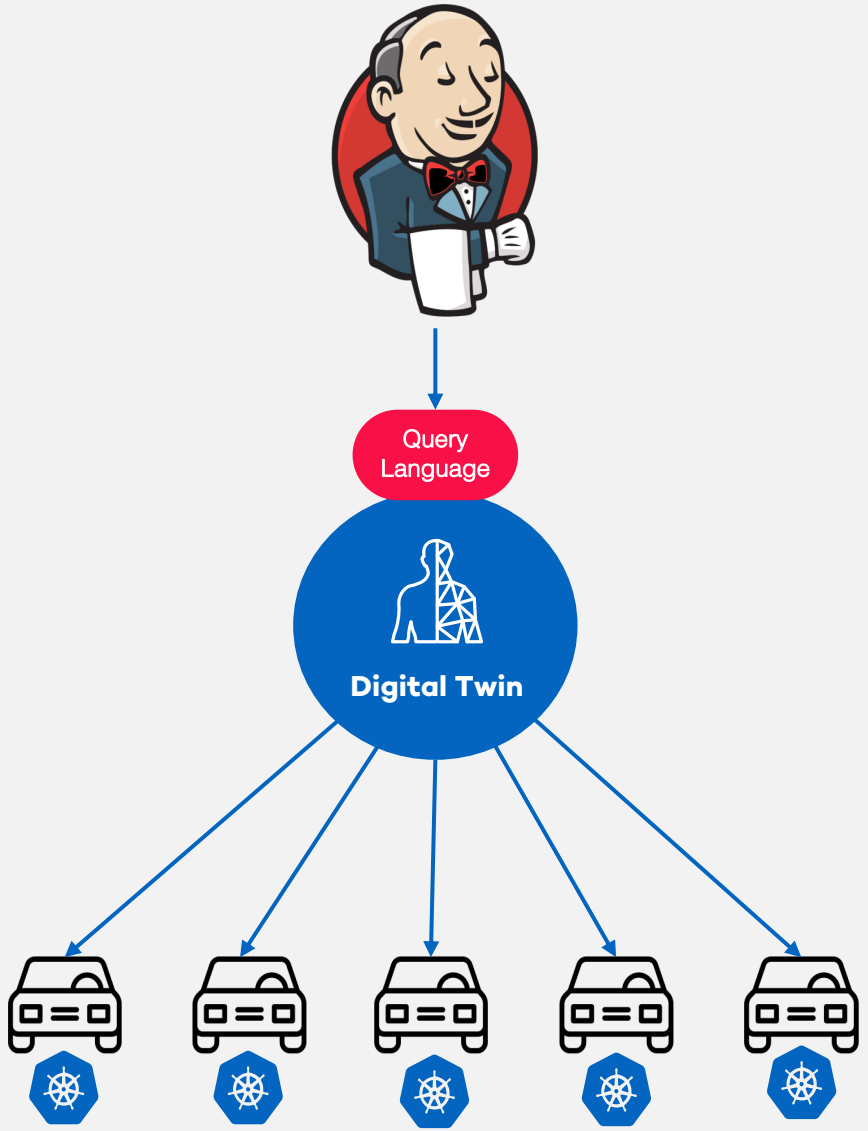
Let's put the deployment flow together

grape up



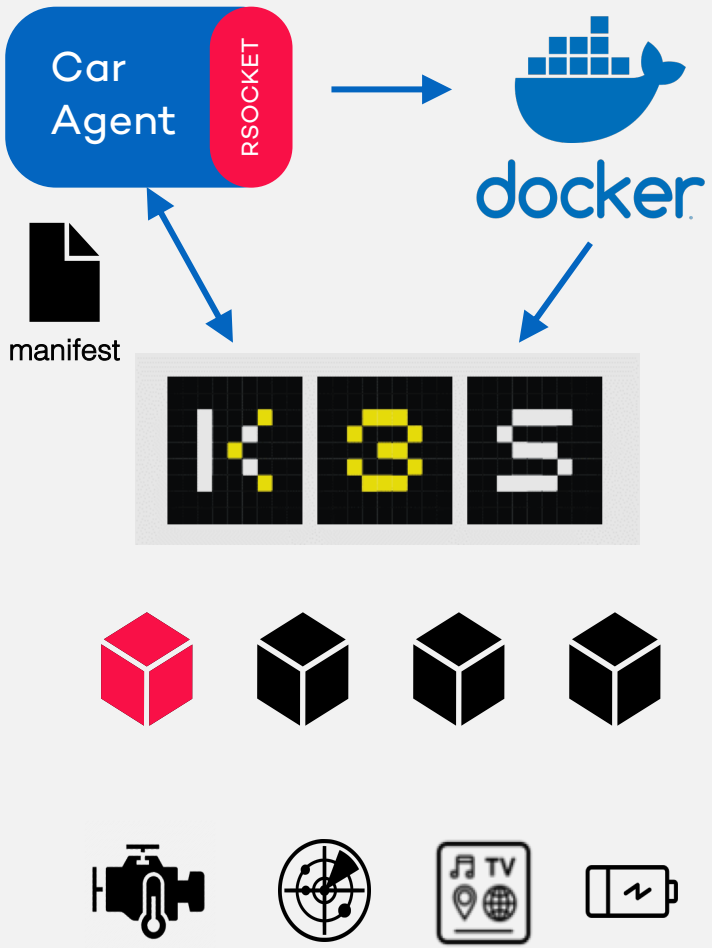
What if I have thousands of cars to update ?

- Nothing changes – cars will act based on the information provided by Digital Twin
- Twin enables **selective deployment**
- **Time to live support**



What about rollback?

- Car has it's own **registry**, so that the switch to previous version of the component is a matter of seconds
- The **car agent is responsible of monitoring** of the deployed components
- Usually two versions of software running in paralel (canary deployment)



Summary

grape up

- Thanks to **μPaas** solutions (like KubeEdge, k3s) we can run “cloud” on the edge. It applies only to large/complex devices, usually we are not fully cloud native.
- **Connectivity** and **number of devices to handle** are main concerns in terms of software delivery to the car – both can be addressed by Digital Twin pattern
- **μPaas and RSocket** helps in unification of the runtime environment and the communication protocols