

Docker and Pharo
@ ZWEIDENKER

We are...



Pierre Chanson



Norbert Hartl



Marcus Denker

ZWEIDENKER

- Founded in 2009 (10 Year party next year!)
- 13 people in Cologne (Germany) and Ho Chi Minh City (Vietnam)
- iPhone, Android, Web, Backend. Pharo!
- You might know the german Call A Bike or Flinkster Car Sharing.

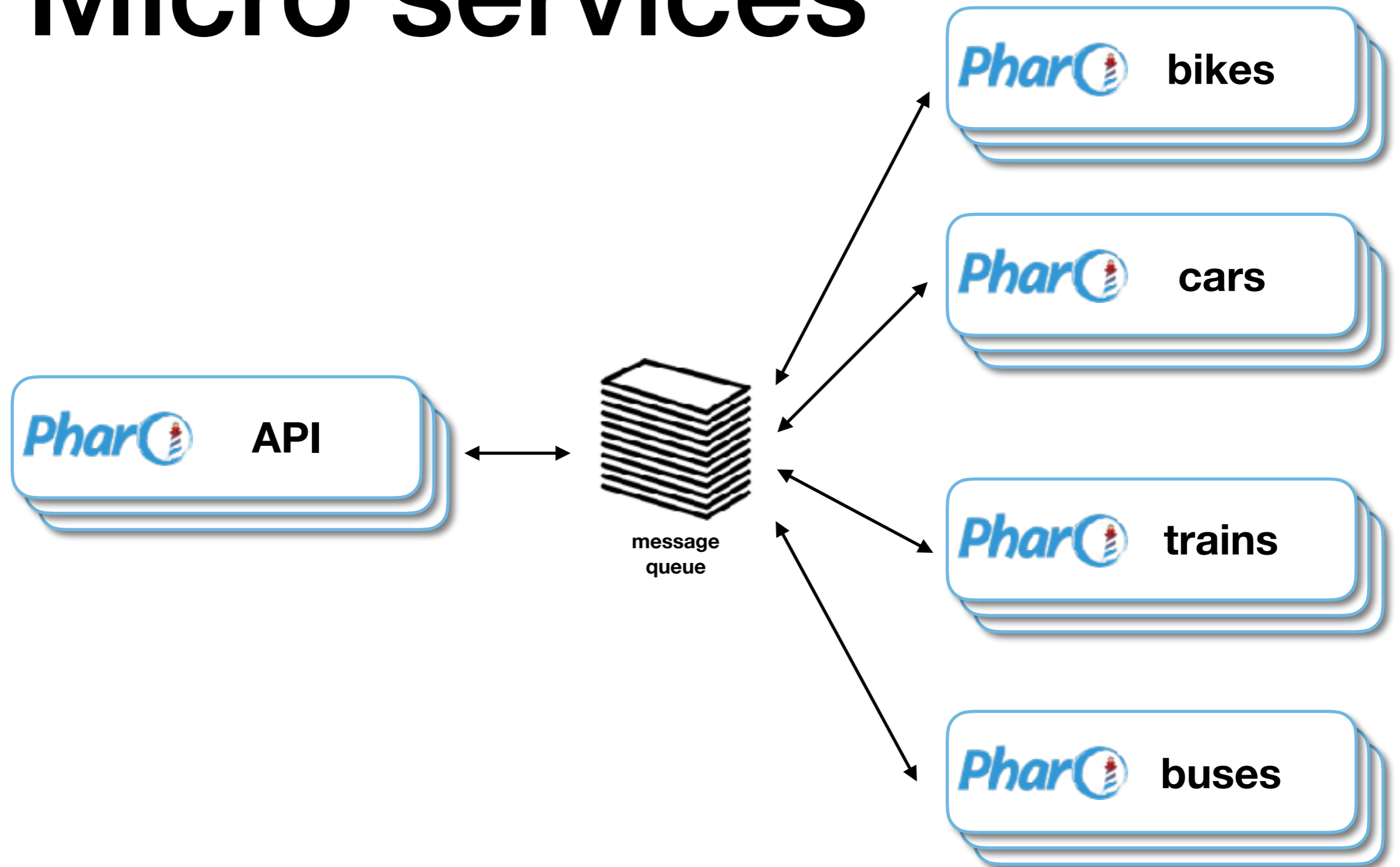
This presentation

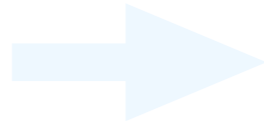
- Server infrastructure
- Application deployment, Monitoring
- Workflow: following one commit step-by-step

The project

- Big project: mobility service for Airbus
- Need for scale: +10k users in a few hours timeframe
- Need not to be killed by complexity

Micro services





Iceberg



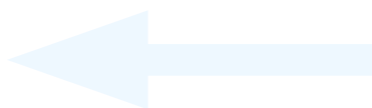
Jenkins



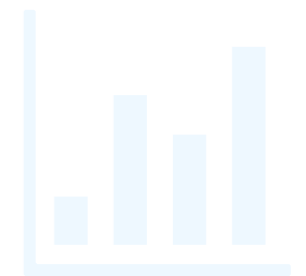
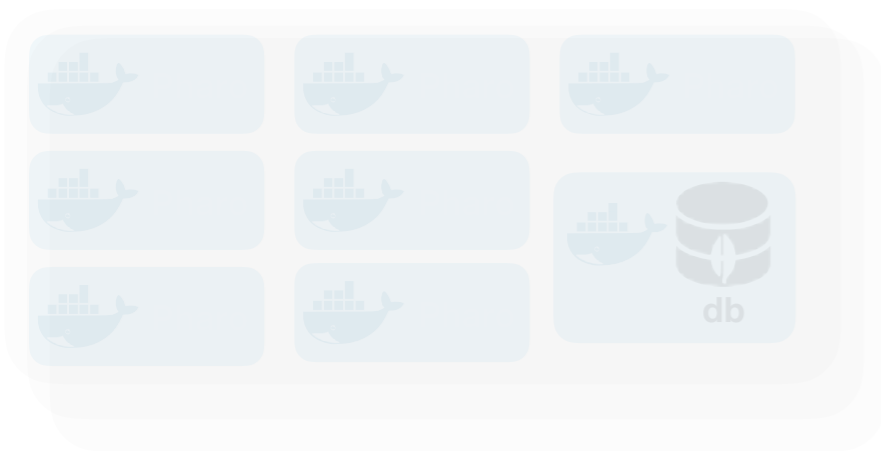
Docker Hub



ANSIBLE



Docker Swarm



Grafana

ID



Regular day of coding at the office...





Regular day of coding at the office...



true become: false

self flag: #bob



Regular day of coding at the office...



true become: false

self flag: #bob

Gotta show this to my boss !



Regular day of coding at the office...

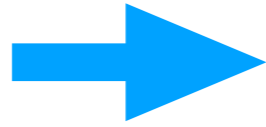


true become: false

self flag: #bob

Gotta show this to my boss !

How ???



Iceberg



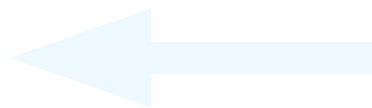
Jenkins



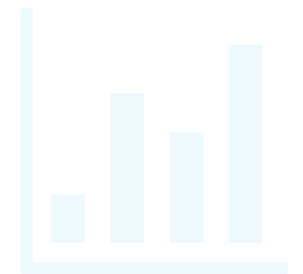
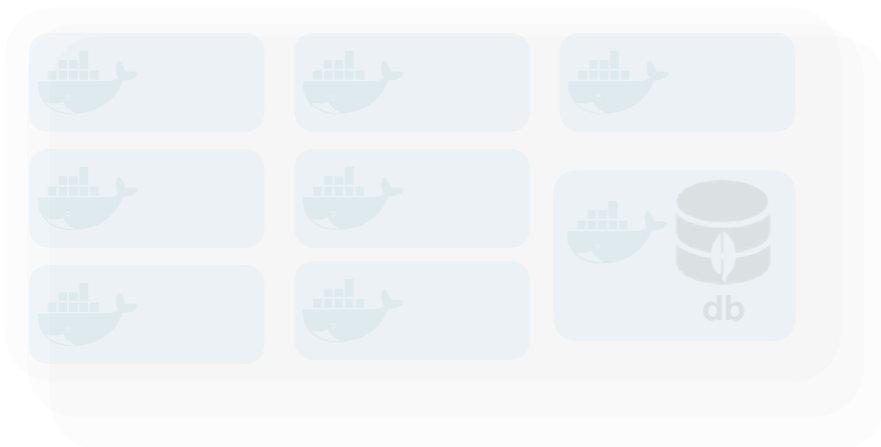
Docker Hub



ANSIBLE



Docker Swarm



Grafana

ID



Git commit

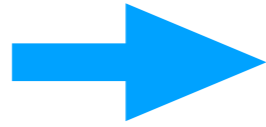
Git push

Pull request

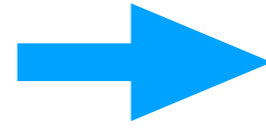


Review...

“Good job” merge



Iceberg



Jenkins



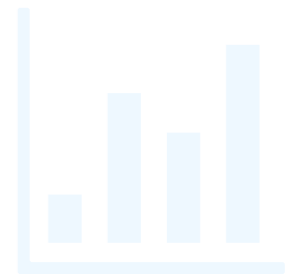
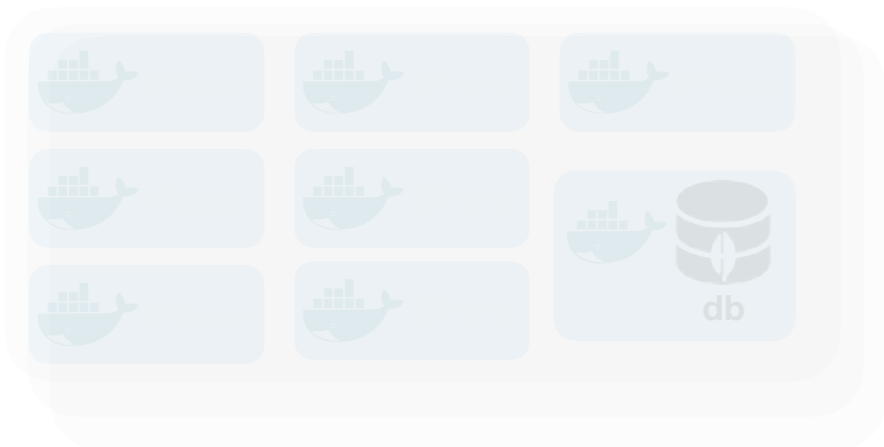
ANSIBLE



Docker Hub



Docker Swarm



Grafana

ID



Wassup ? (every 10 mins)

**Alright, pull
run tests**

docker ...



What is Docker?

- Think about it as a Smalltalk image, but for Linux
- Docker Image is build from a description (Dockerfile)
- Then we can run it (once or multiple times)

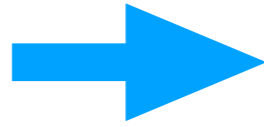


Wassup ? (every 10 mins)

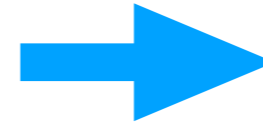
**Alright, pull
run tests**

**docker-compose build
docker-compose push**





Iceberg



Jenkins



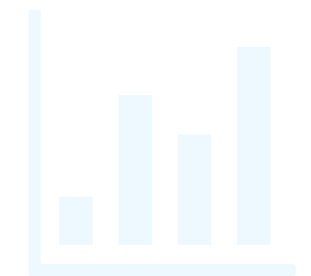
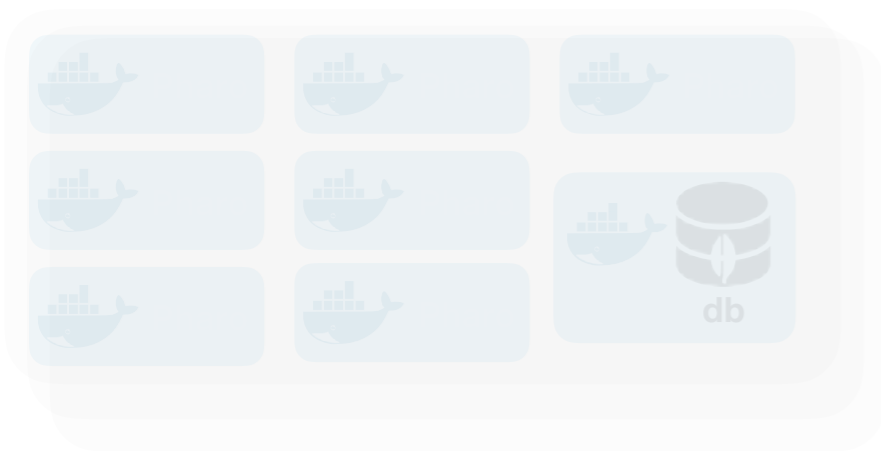
Docker Hub



ANSIBLE

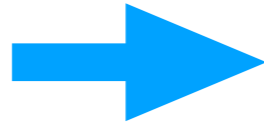


Docker Swarm

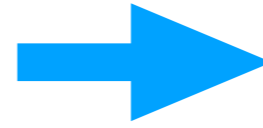


Grafana

ID



Iceberg



Jenkins



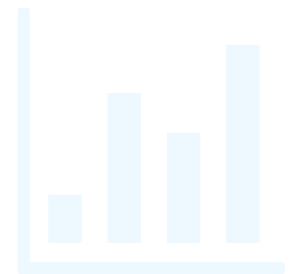
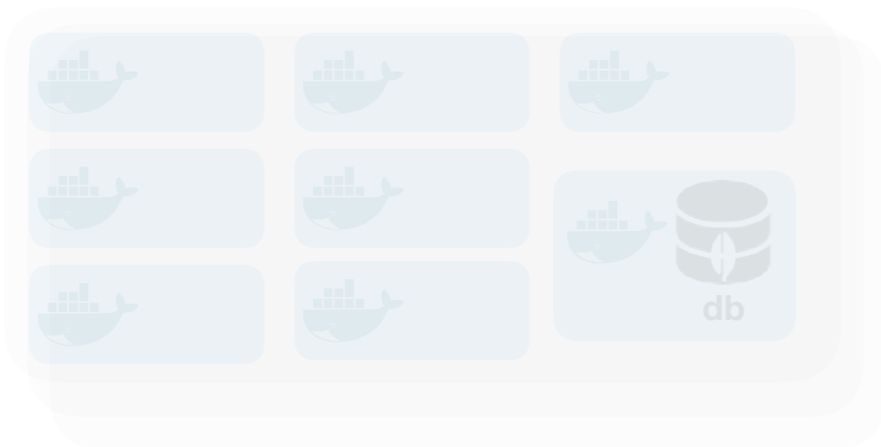
Docker Hub



ANSIBLE



Docker Swarm



Grafana

ID

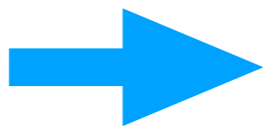
Ansible: Orchestration

- We need to start lots of stuff (Pharo, Webserver, MessageQueue, Database...)
- Ansible is a way to script typical unix admin things
- Both used to setup the Linux machines and to start Docker

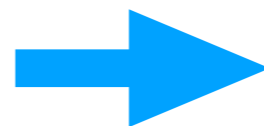


ansible-playbook -i myInventory myYml.yml

(there is nothing worse than yaml!)



Iceberg



Jenkins



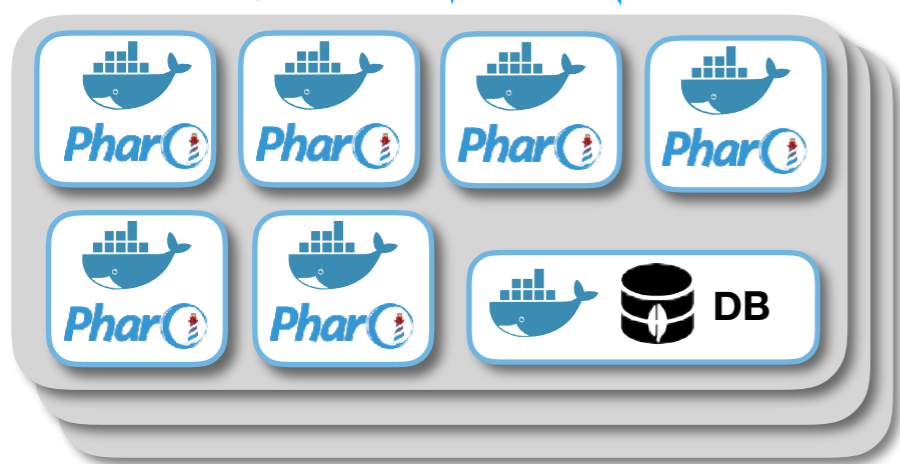
Docker Hub



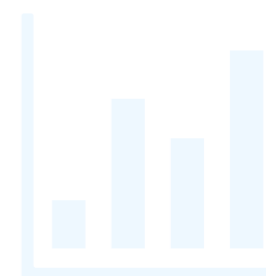
ANSIBLE



Docker Swarm

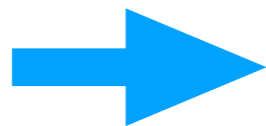


Grafana

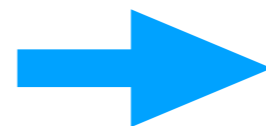


Docker: Swarm

- We want to run on multiple machines
- Swarm allows to run “services” which scale to multiple docker containers
- Manages Distribution, restart...



Iceberg



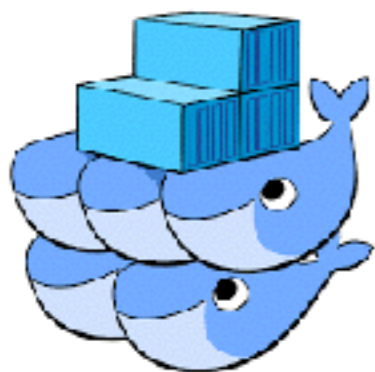
Jenkins



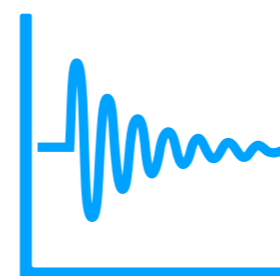
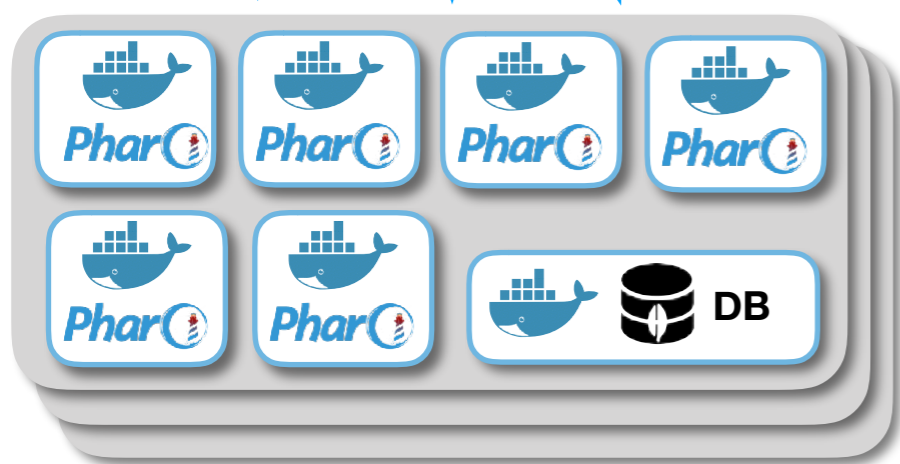
Docker Hub



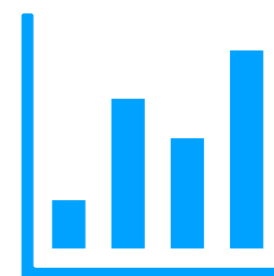
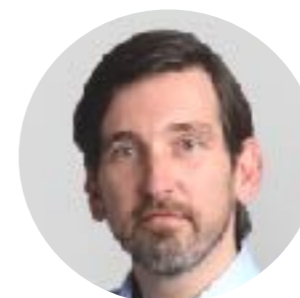
ANSIBLE



Docker Swarm



Grafana

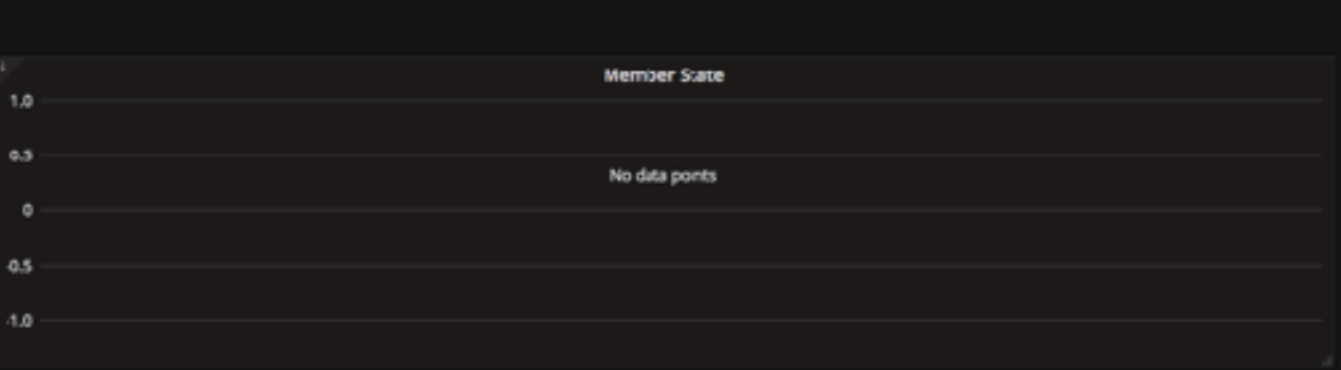
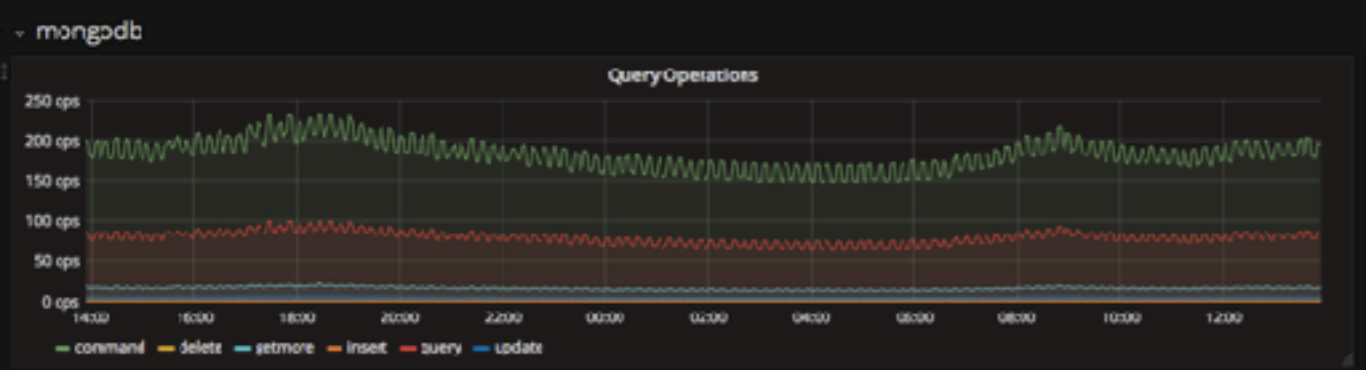
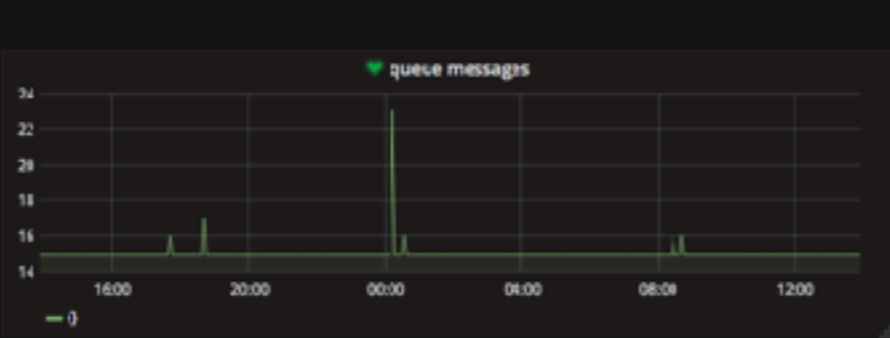
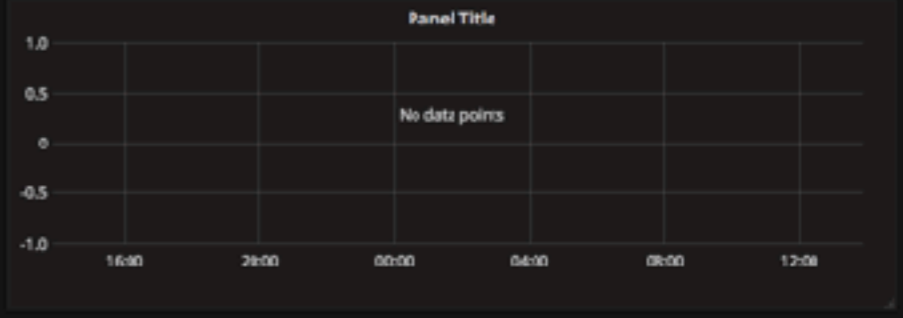
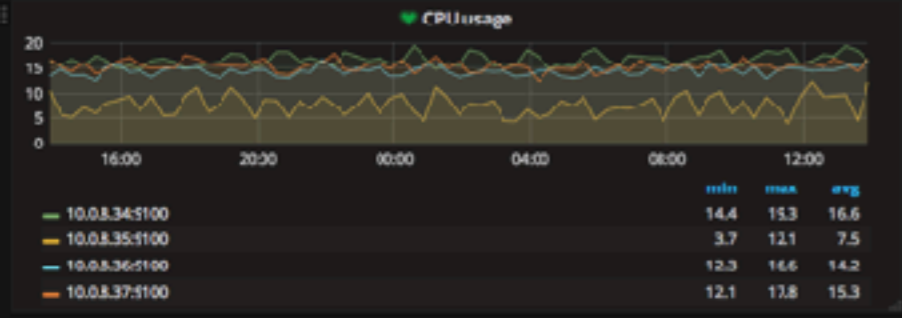
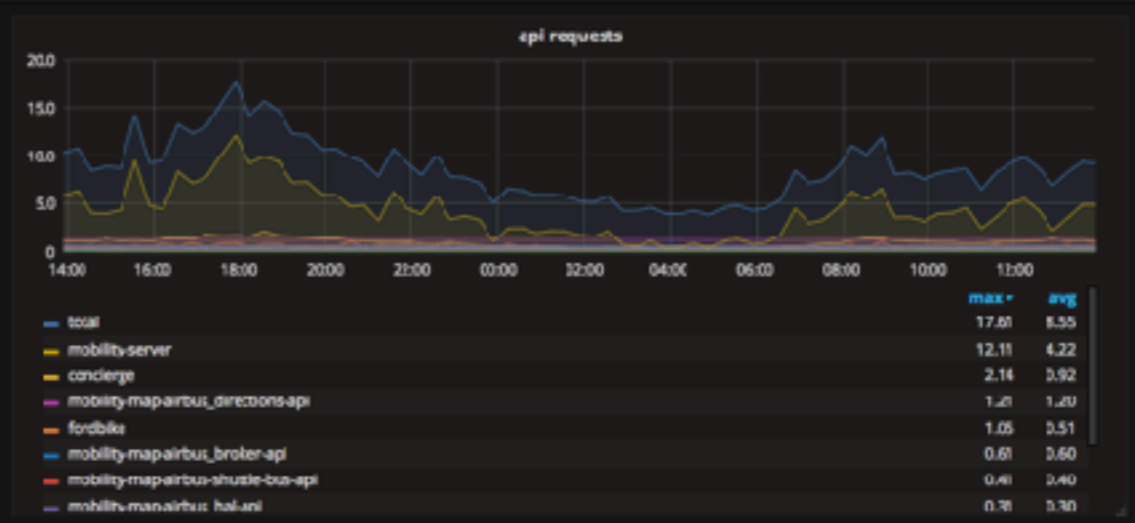
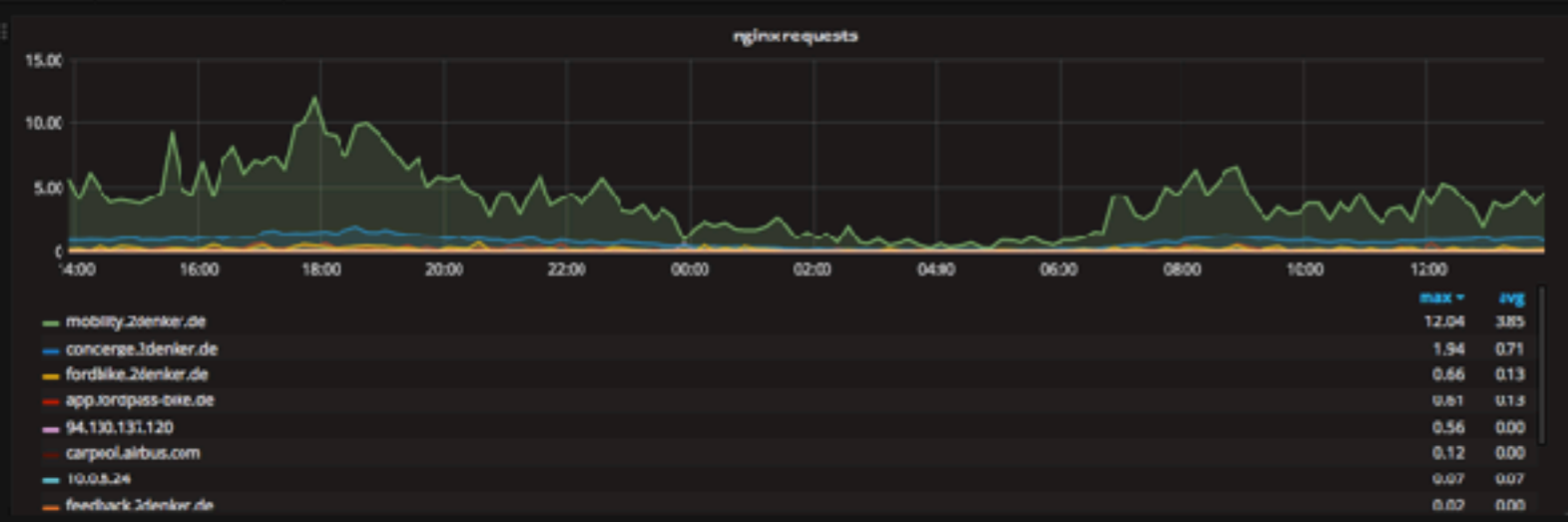


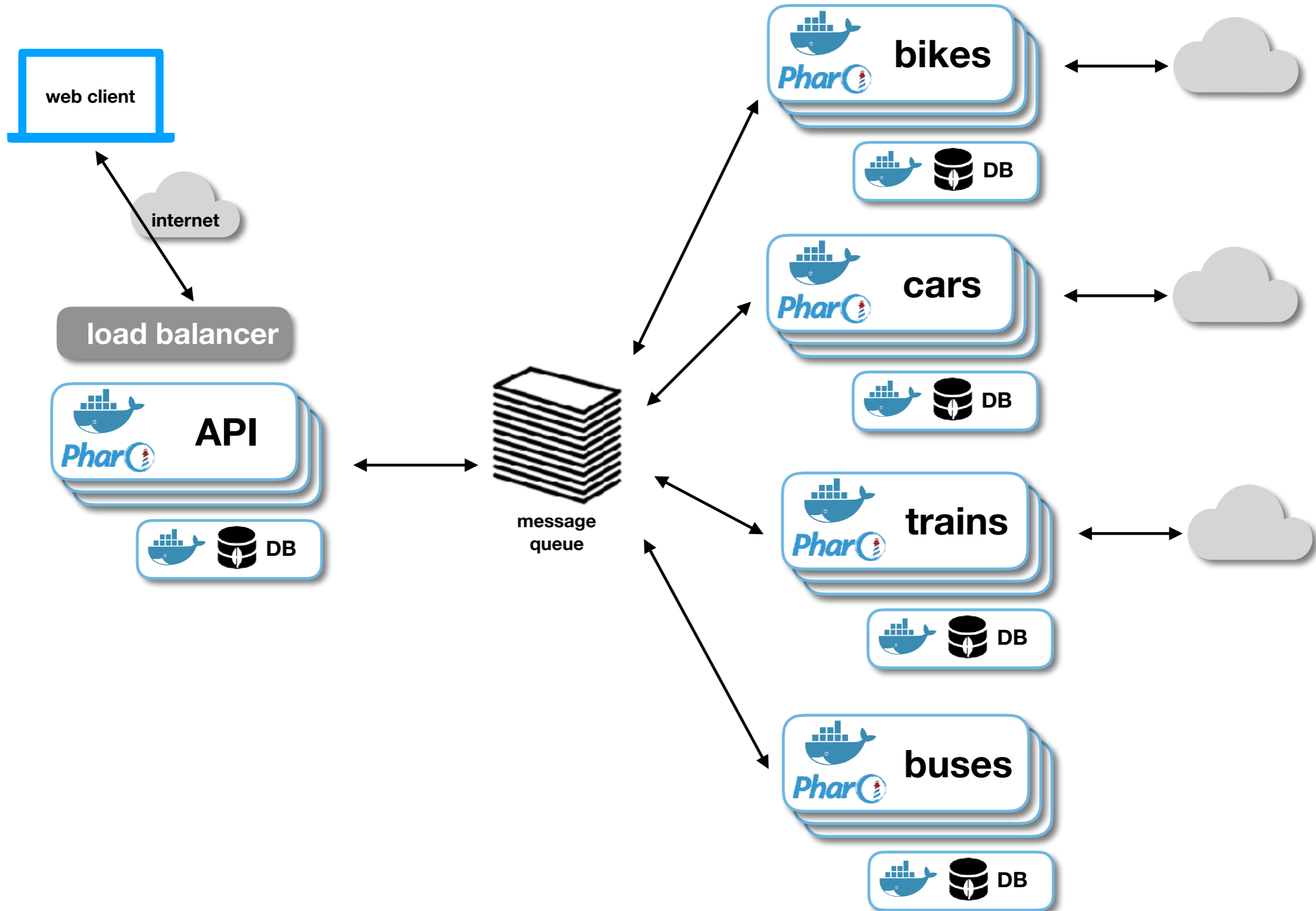
Monitoring

- Lots of machines (virtual and real)
- Lot of different software (database, web, message queue, lots Pharo images)
- How do we look at it?

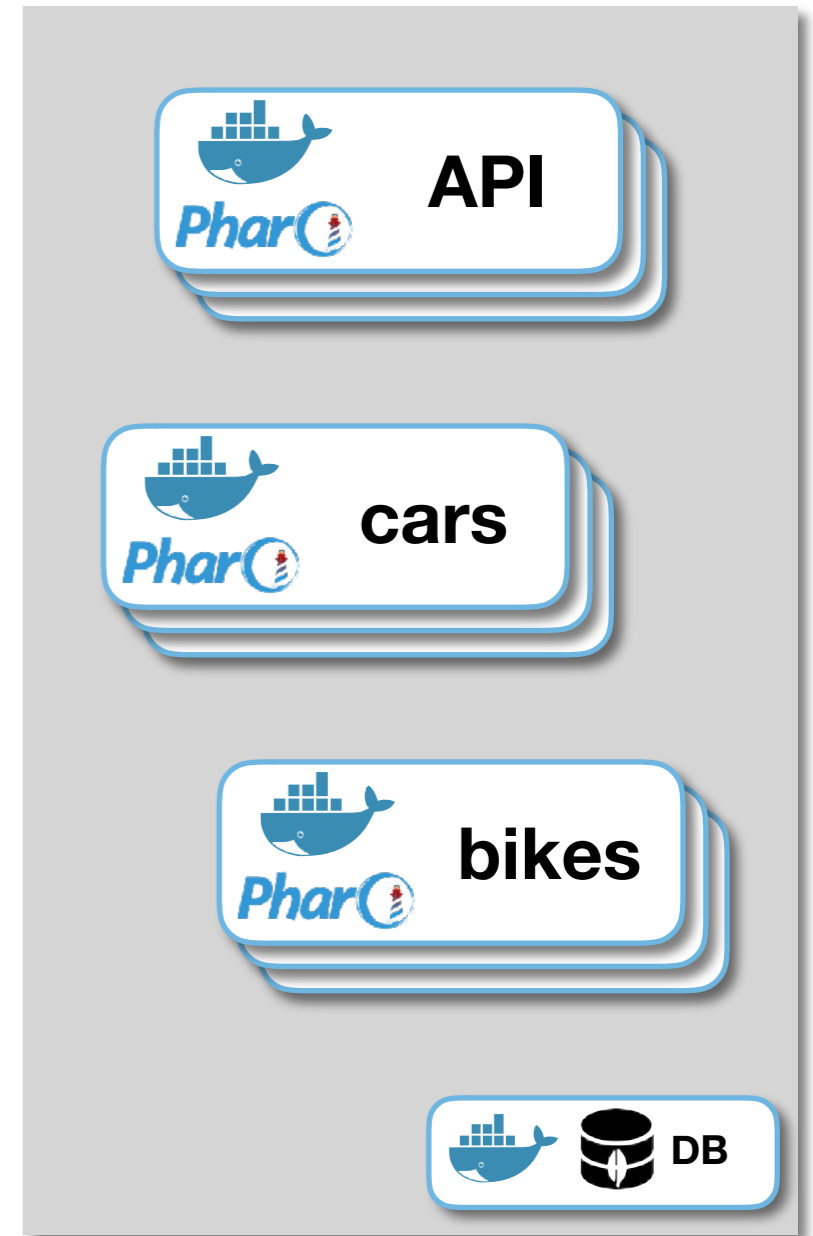
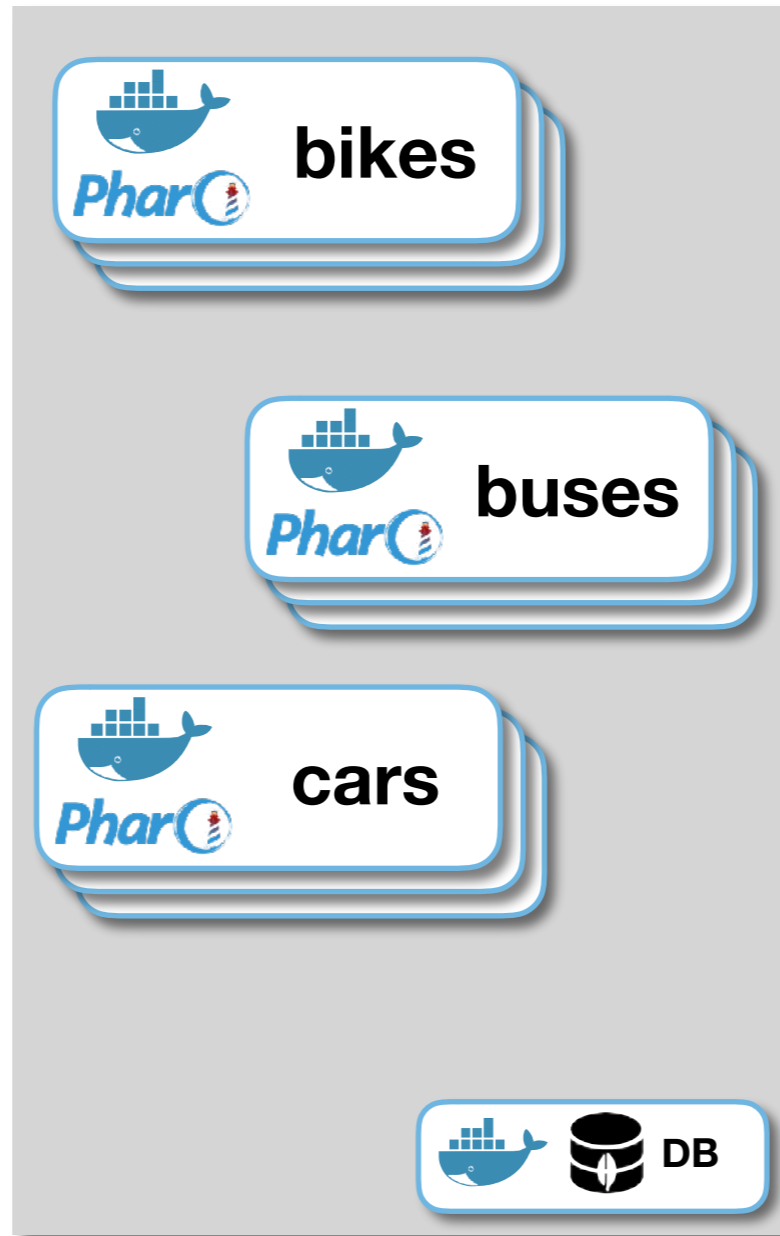
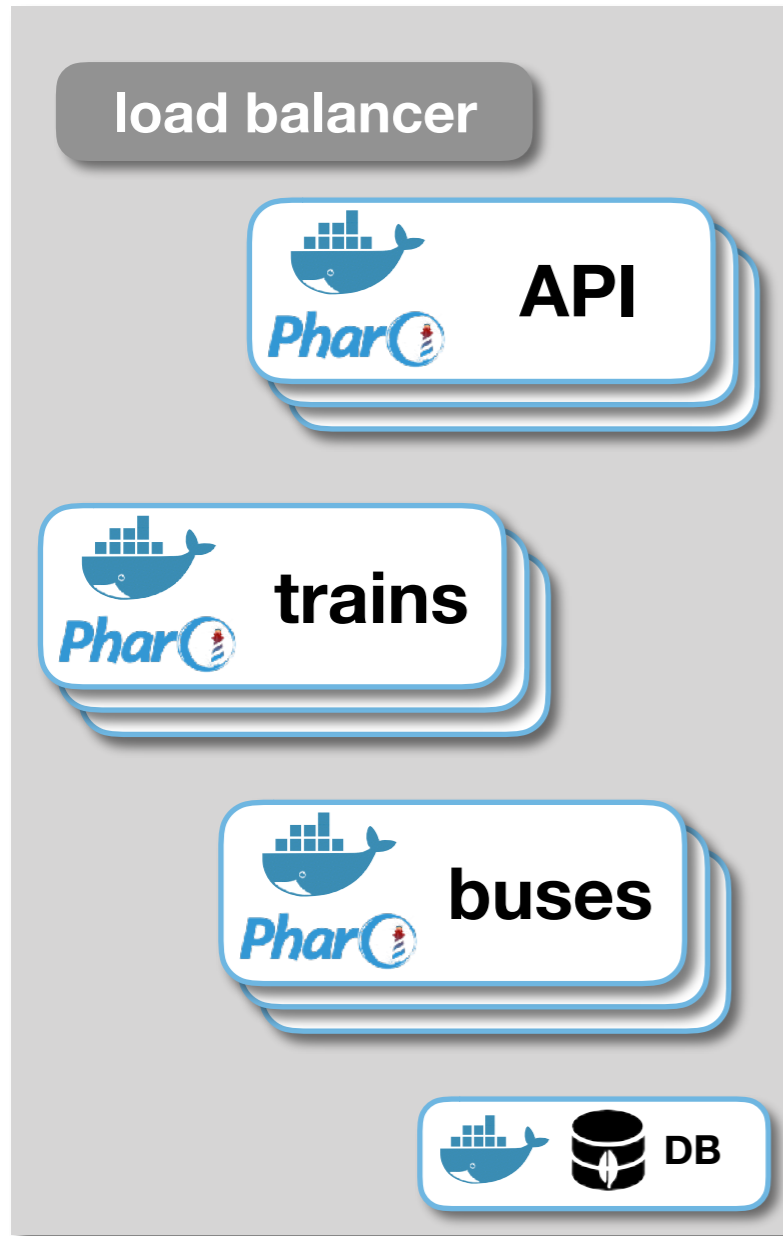
Monitoring

- Grafana
 - Dashboard
 - Alarms: Get Notifications via Mail and in Slack
- Docker images provide data via REST API
 - Pharo GC, Zinc Requests





Physical servers



And that x 3

- We have 3 Swarms:
 - Alpha (development) **+100 Pharo images overall in our cluster**
 - Beta (testing)
 - Production
- And you can run the whole setup on your laptop, too!

Much more...

Pharo Library for Docker Swarm (work in progress)

The screenshot shows a Pharo Playground window with a code editor on the left and a visual representation of a Docker Swarm cluster on the right. The code in the editor defines a library for interacting with Docker Swarm, including methods for listing nodes, tasks, and containers, and for inspecting specific tasks and containers. The visual representation shows a grid of nodes, each with a unique ID and a status indicator.

```
specFile := 'pharo-local/iceberg/zweidenker/DockerEngine/openapi3.json' asFileReference.
openApi := OpenAPI fromString: specFile readStream upToEnd.
docker := DockerEngineClient new
  openApi: openApi;
  baseUrl: 'http://localhost:2376'.

nodes := docker nodes.
docker tasksInNode: nodes third ID.
containers := docker containers.
docker inspectContainers: containers anyOne.

taskListItems := docker tasks first.
task := docker taskWithId: taskListItems ID.
containerId := task status ContainerStatus ContainerID.

docker containerWithId: containerId.

view := RView new.
elements := (RTBox new color: Color lightGray) elementsOn: docker nodes.
view addAll: elements.
RINest new
  for: elements
  add: | :group :model | | comp |
    comp := RTCompositeShape
      with: (RTBox new width: 25; height: 18; color: Color lightBlue)
      with: (RTLabel new height: 3; text: [: x | x ID first: 18 ]).
    el := comp elementsOn: (docker tasksInNode: model ID) .
    group addAll: el.
  RIGridLayout new gapSize: 2; applyOn: group ].
RTFlowLayout new on: elements.
elements @ RTPopup @ RTDraggable.
view
```

The visual representation shows a grid of nodes, each with a unique ID and a status indicator. The nodes are arranged in a grid, with some nodes having a status indicator (a small square) next to their ID. The nodes are arranged in a grid, with some nodes having a status indicator (a small square) next to their ID.

Much more...

- Lots of in-house libs
 - Visualise Docker Swarm with **Roassal** (first work)
 - Docker **Health Check** from Pharo
 - Server for and Pharo app for **Fuel** Stack Traces
 - **DockerEngine + OpenAPI**

Tell us what you want to see at the **Show Us Your Project !**