PGS SOFTWARE
Part of Xebia

Od zera do kontenera

Po co komu docker?

Michał Borkowski

mborkowski@pgs-soft.com

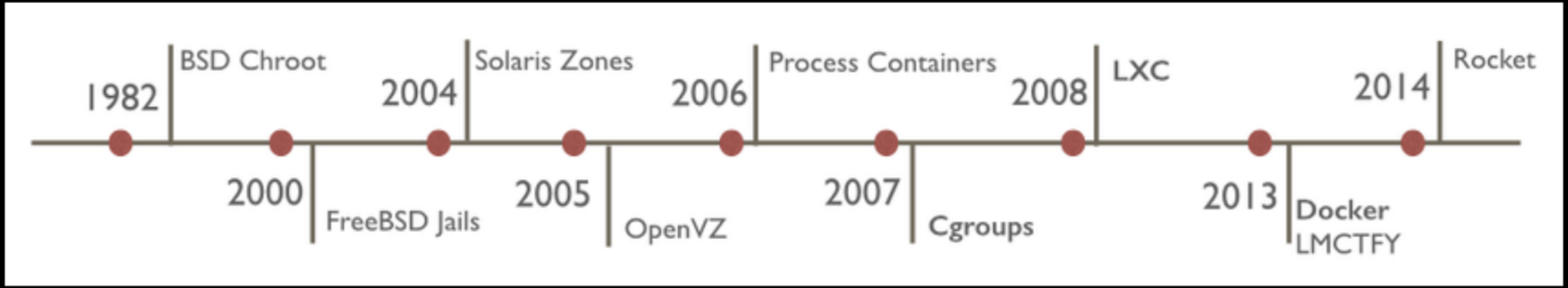
https://norasoft.eu/talks/container-magic/

O czym dzisiaj?

- czym jest docker?
- kontenery bez dockera i obrazu
- dlaczego używamy obrazów?

Czym jest docker?

Od kiedy wirtualizujemy?



Docker jest wszędzie

Google and Containers

Everything at Google runs in a container.

Internal usage:

- Resource isolation and predictability
- Quality of Services
  - batch vs. latency sensitive serving
- Overcommitment (not for GCE)
- Resource Accounting

We start over 2 billion containers per week.

Google Cloud Platform

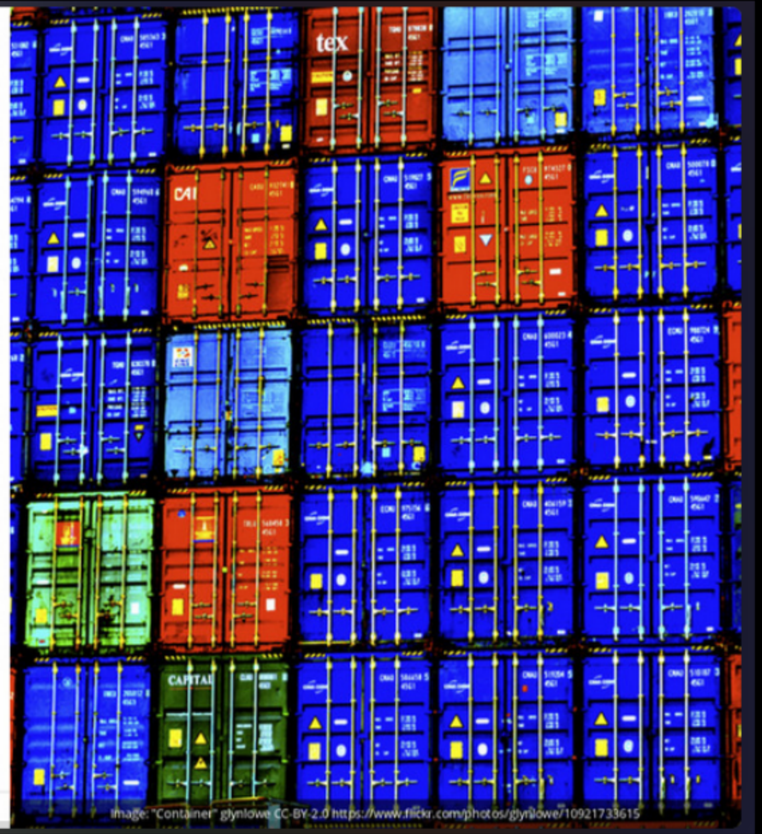
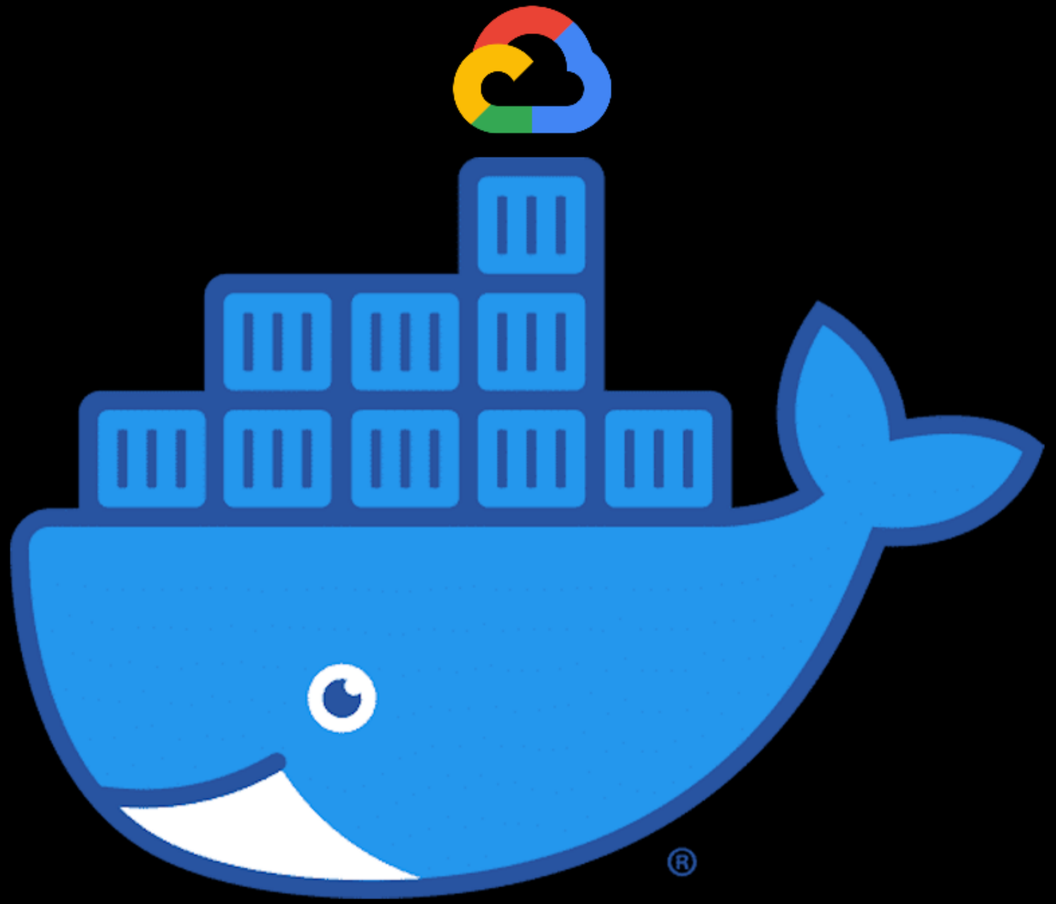


image: "Container" glynlowe CC-BY-2.0 https://www.flickr.com/photos/glynlowe/10921733615

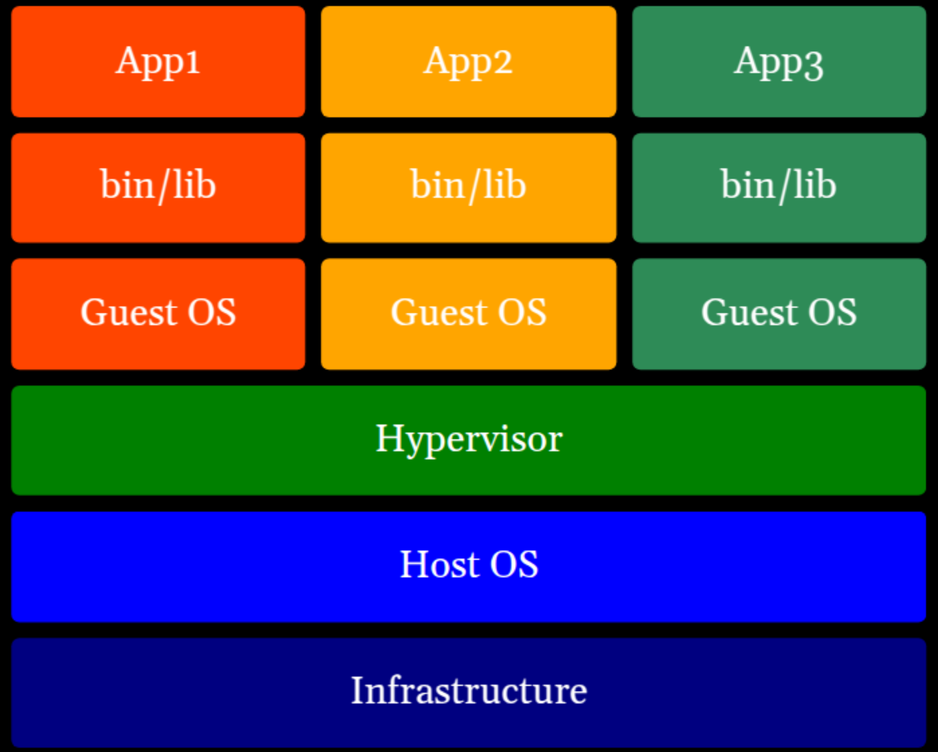
Docker jest wszędzie

Docker jest wszędzie

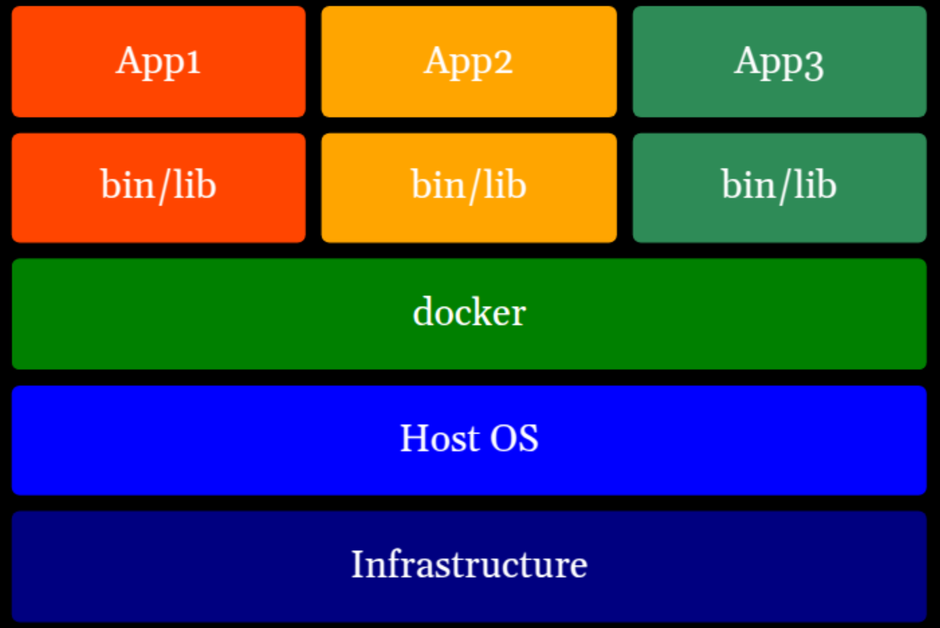


Docker jest wszędzie

Co dały nam kontenery?

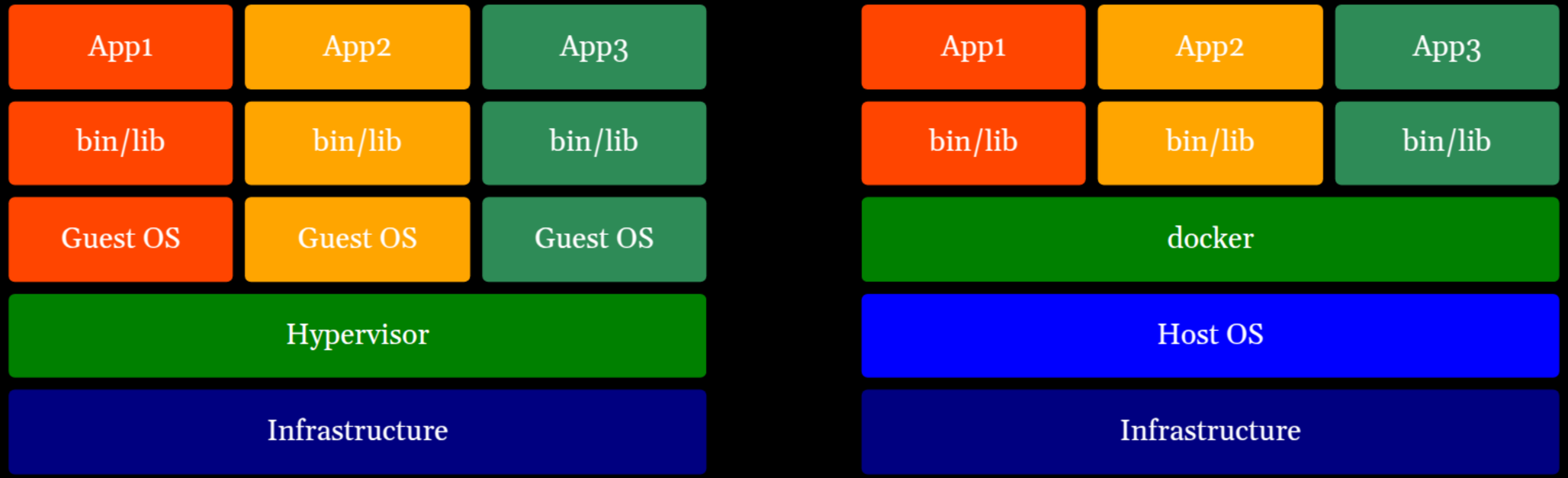


Virtual Machines



Containers

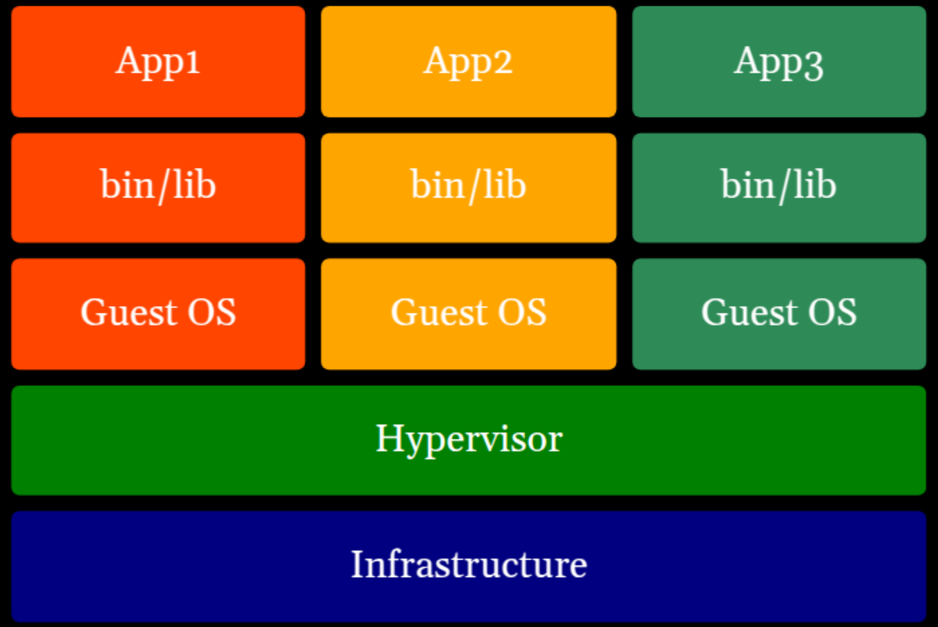
Co dały nam kontenery?



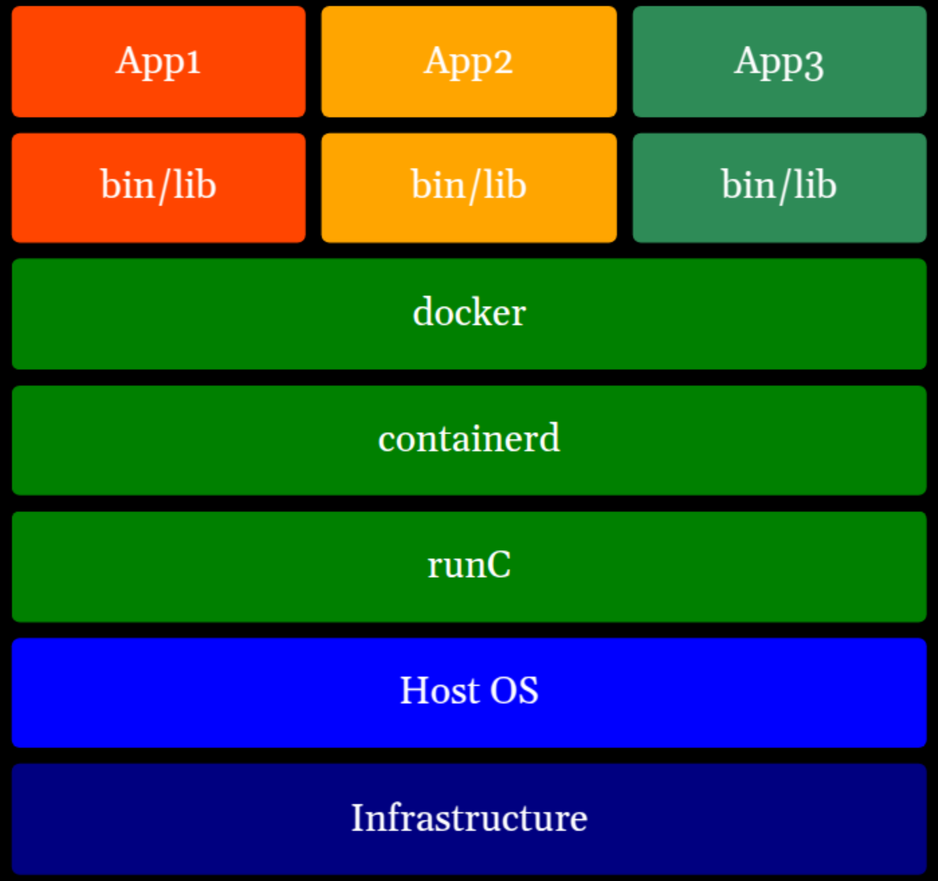
Virtual Machines

Containers

Co dały nam kontenery?

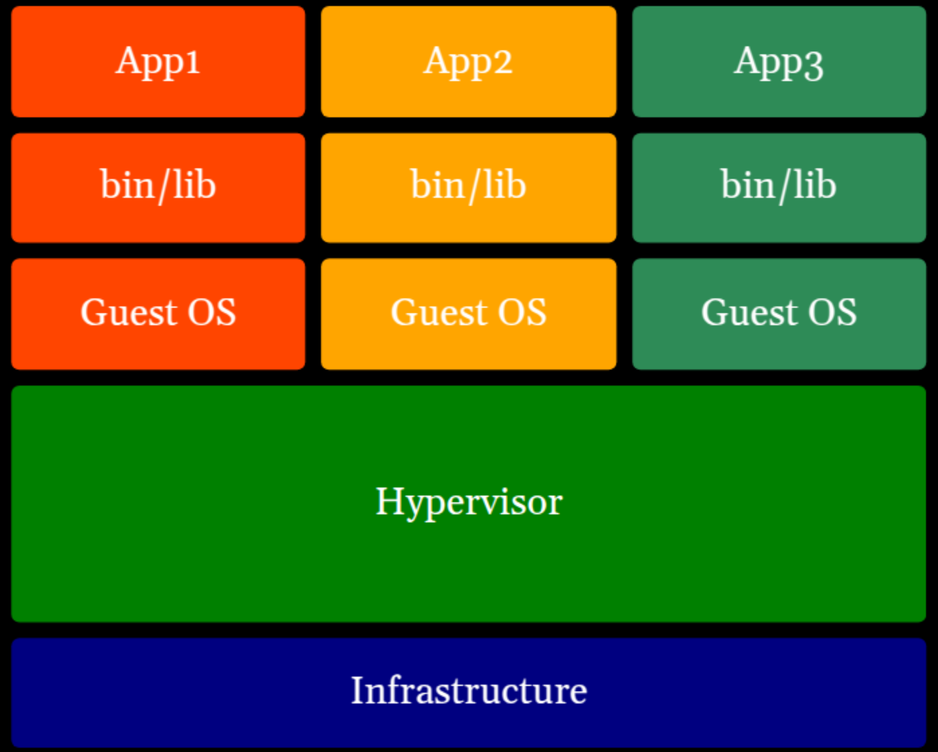


Virtual Machines

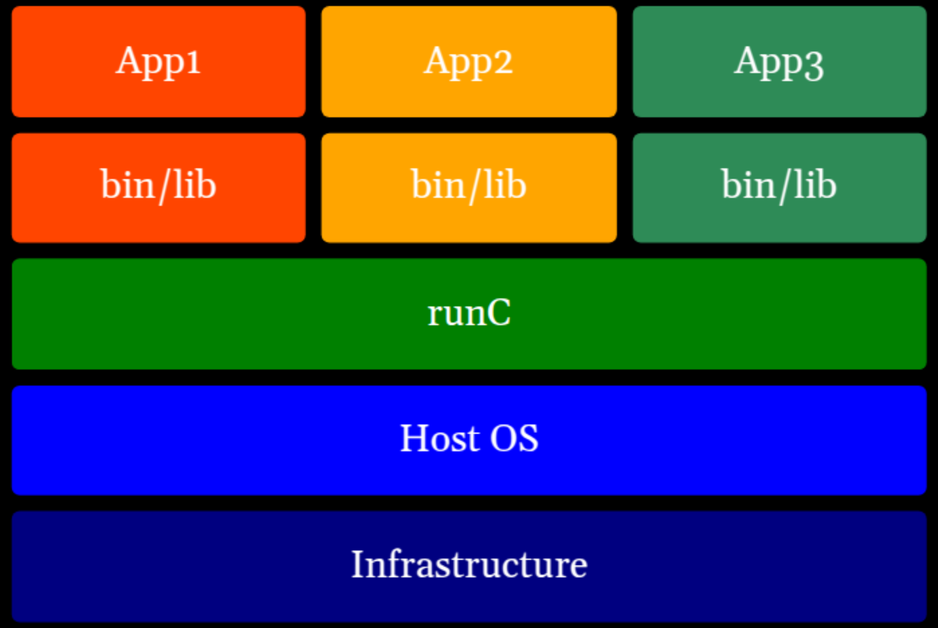


Containers

Co dały nam kontenery?

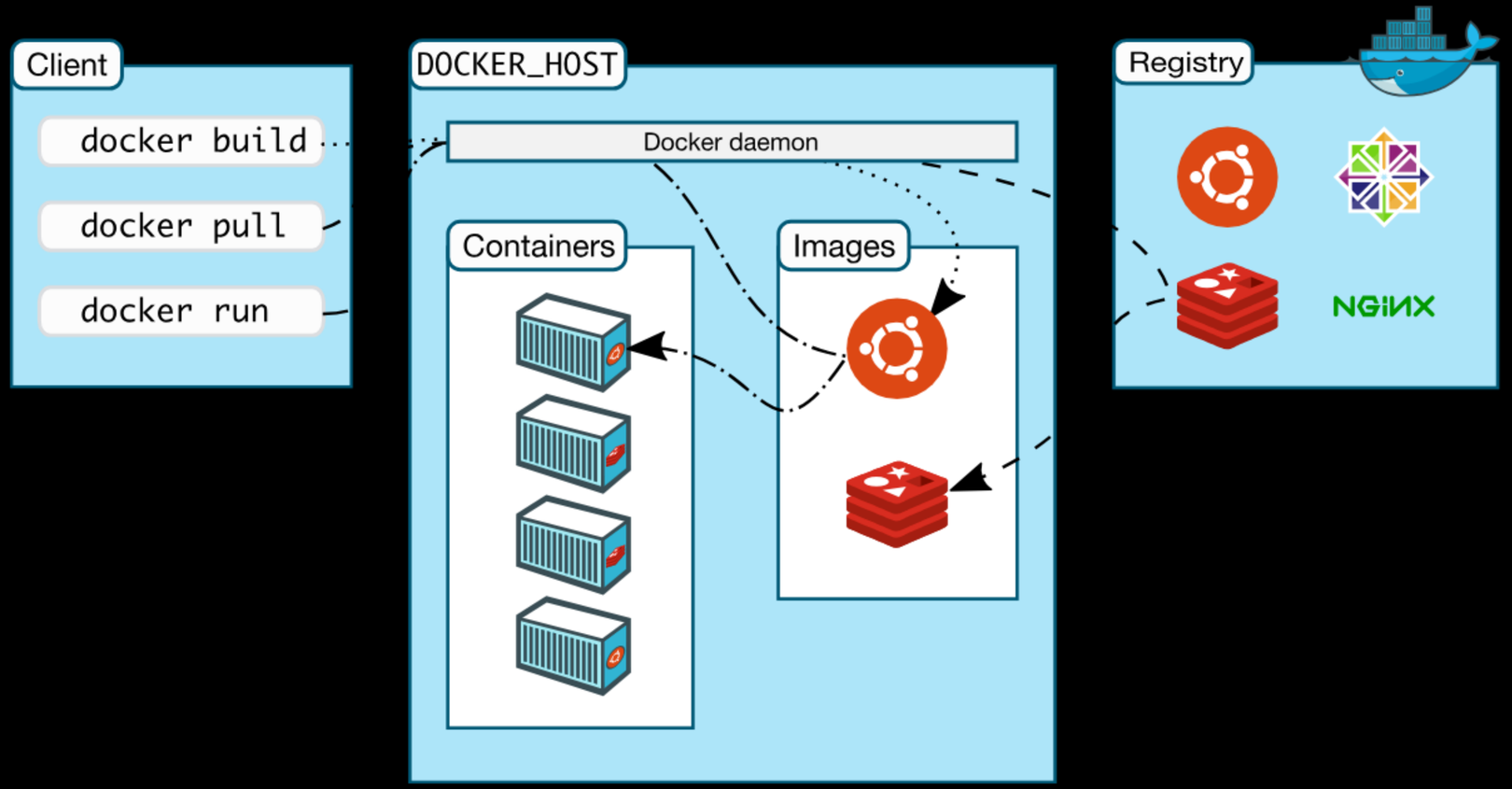


Virtual Machines



Containers

Komponenty dockera



From docker documentation

Kontenery

Kontenery

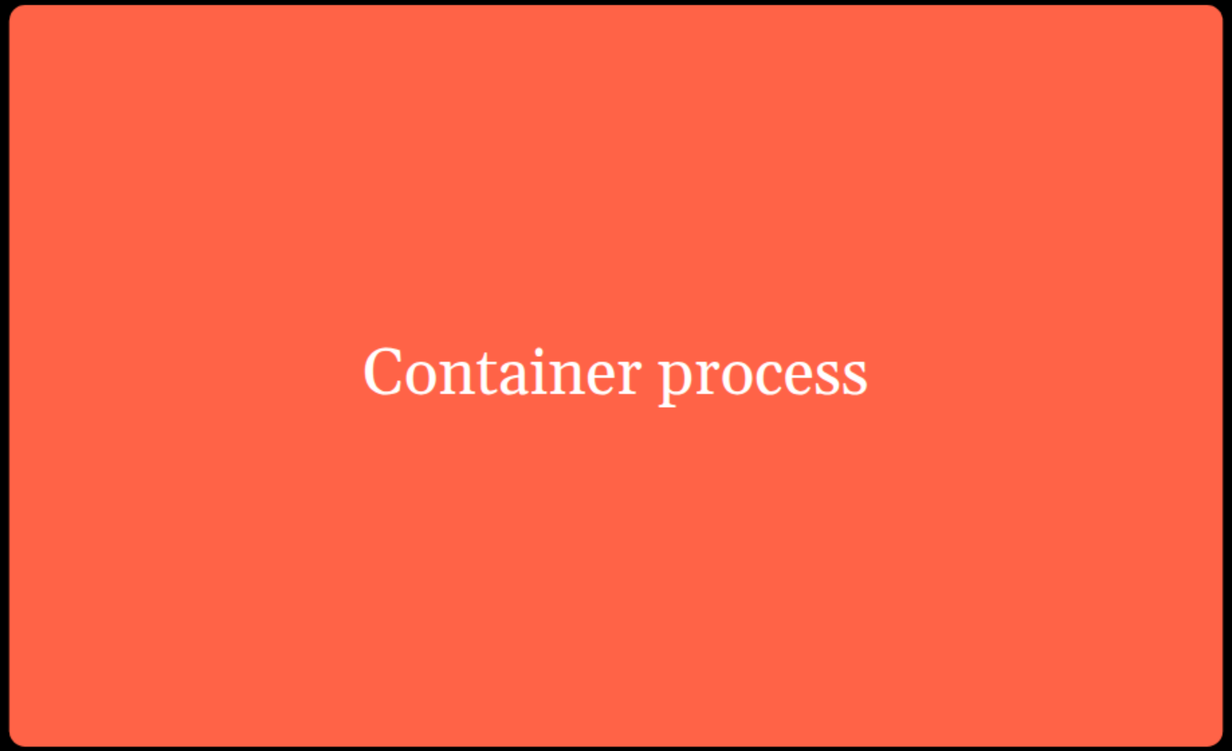
Containers are simply isolated
and restricted Linux processes.
Ivan Velichko - 2020

Kontenery

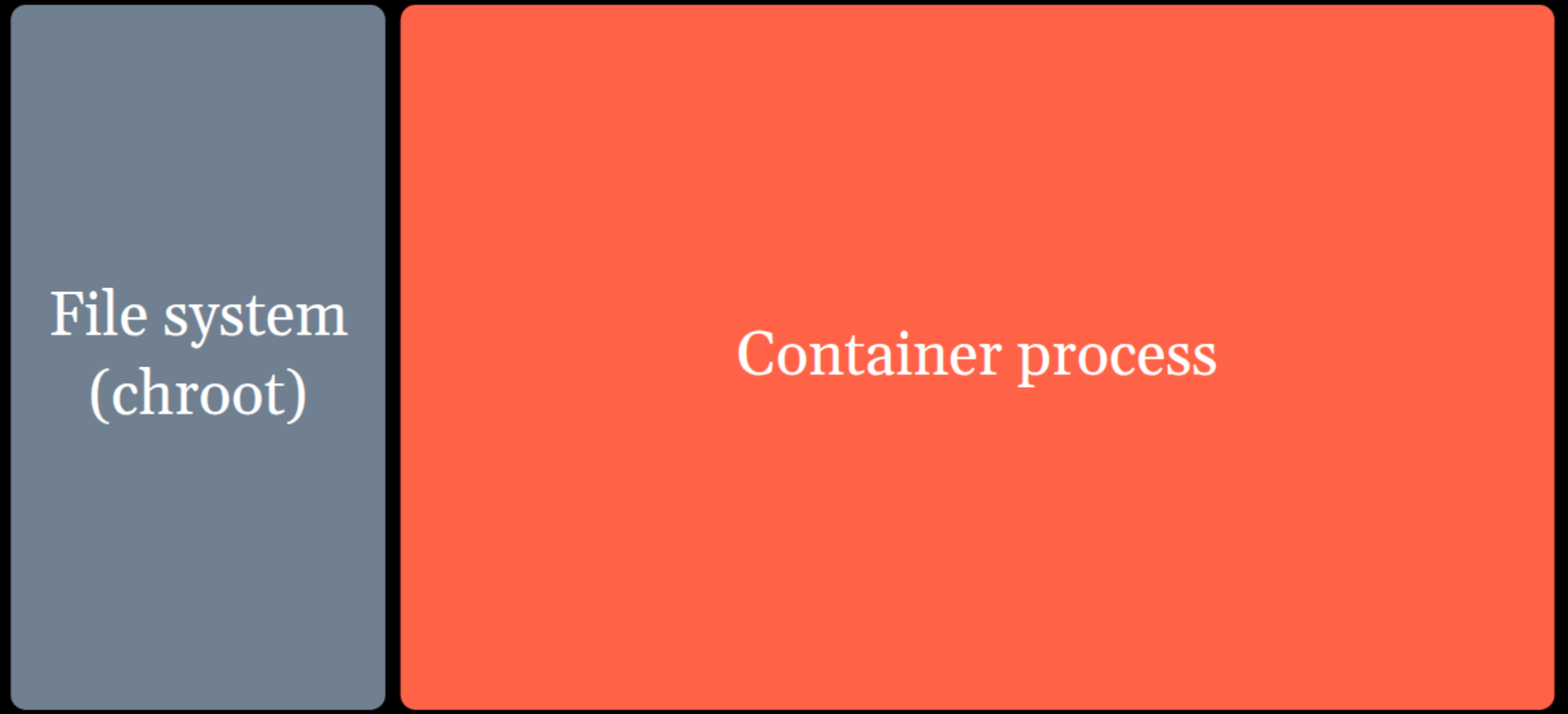
Containers are simply isolated and restricted Linux processes.
Ivan Velichko - 2020

Containers Aren't Linux Processes
Ivan Velichko - 2021

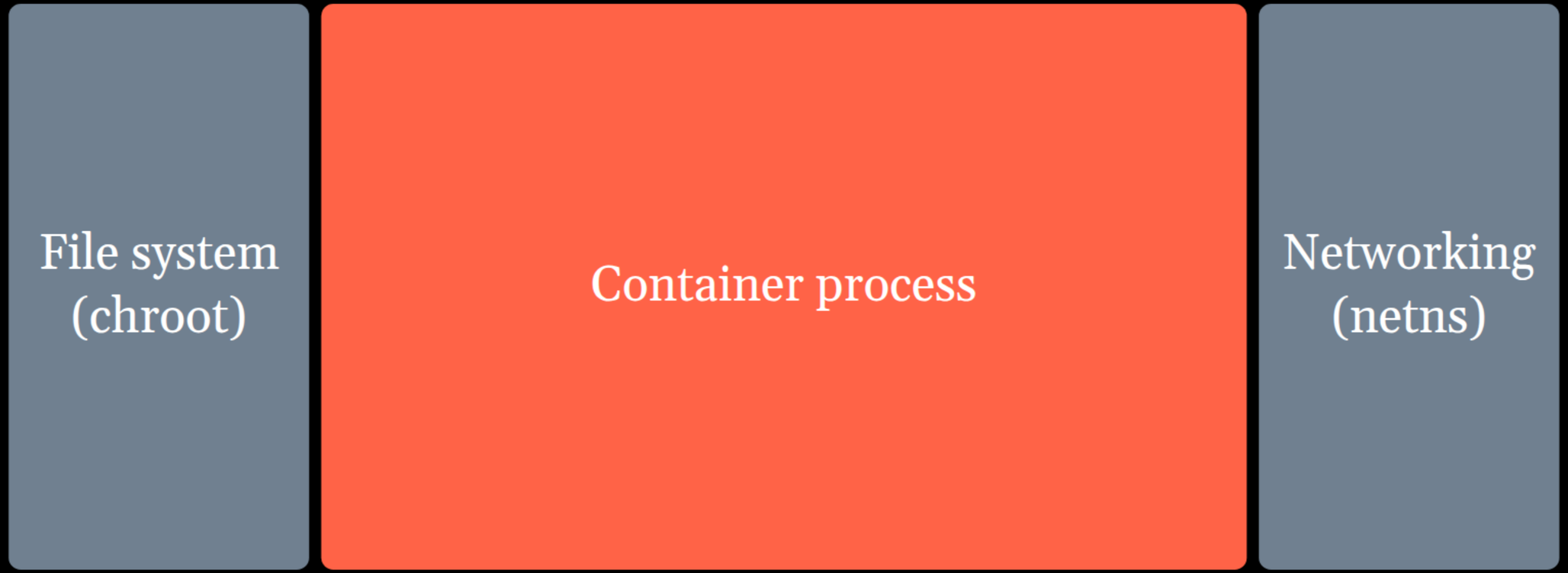
Wnętrze kontenera



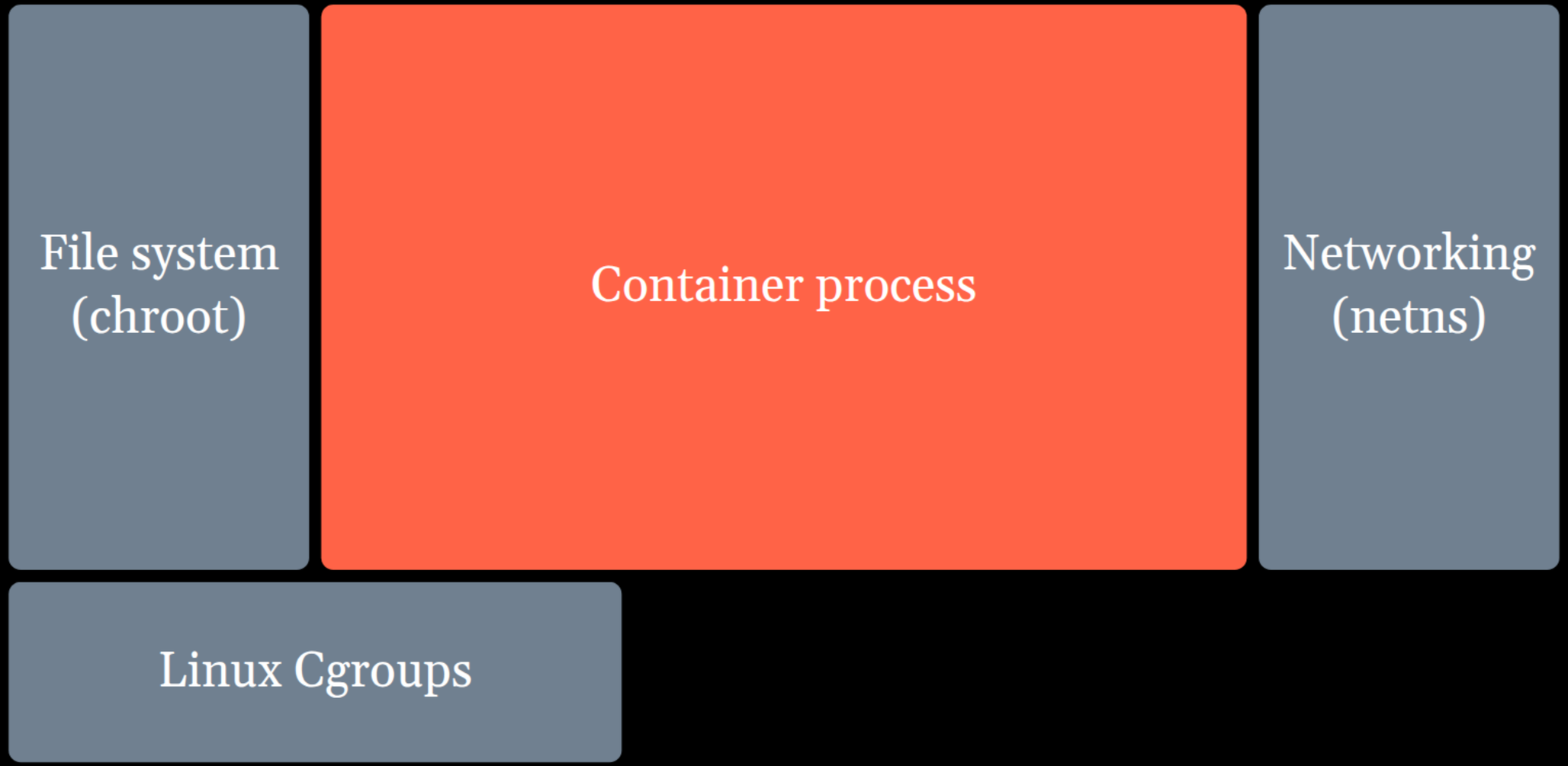
Wnętrze kontenera



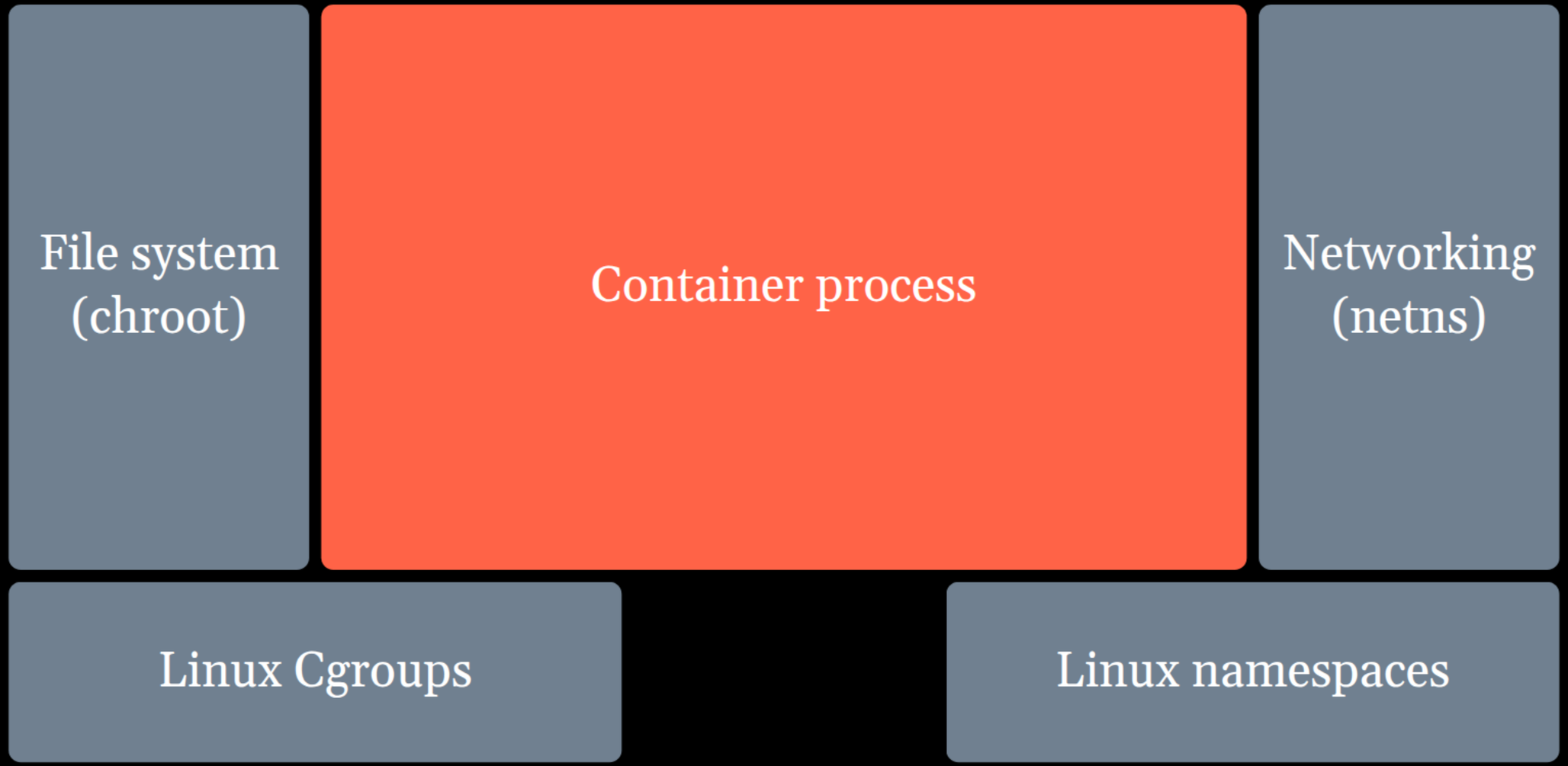
Wnętrze kontenera



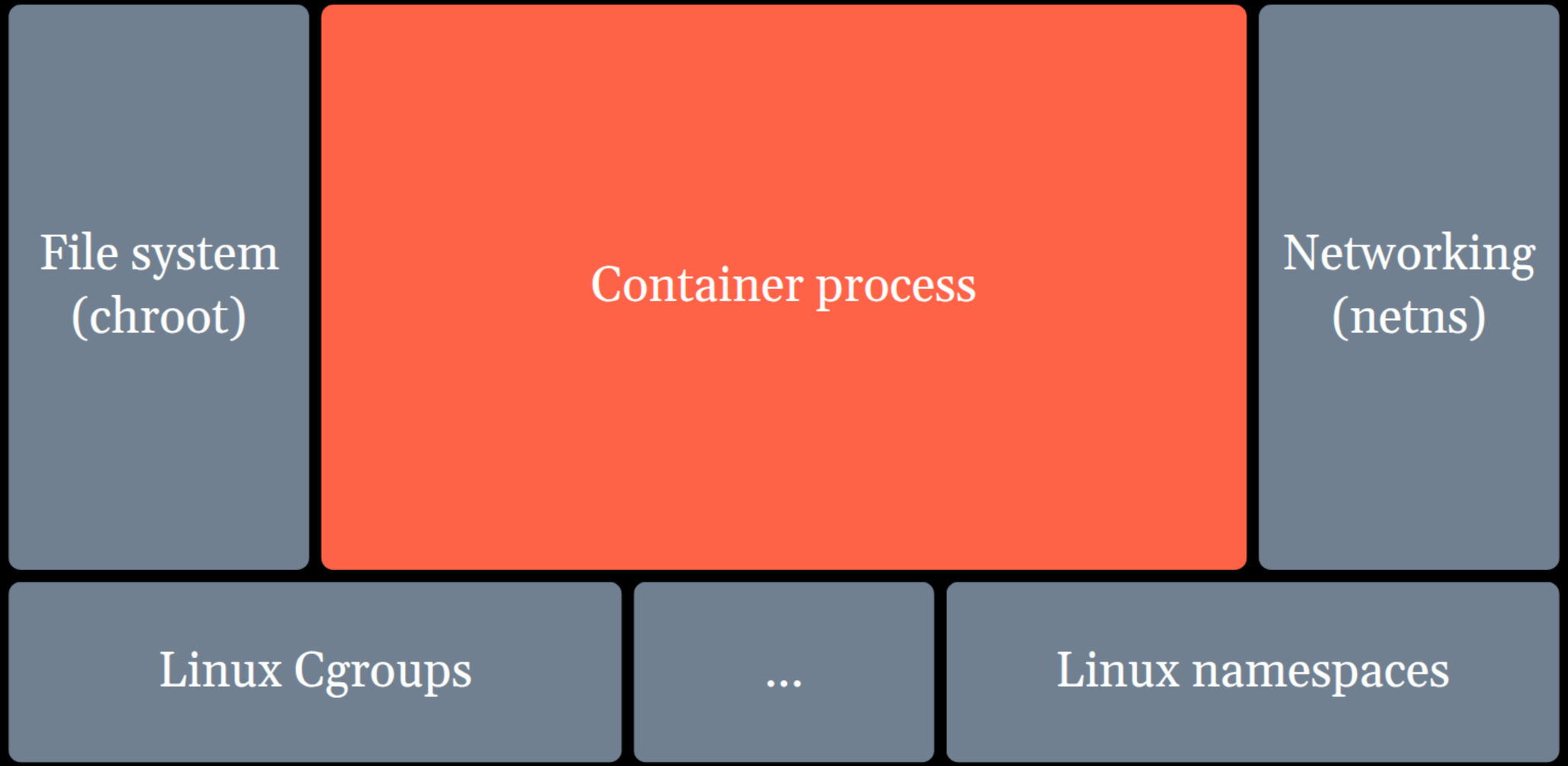
Wnętrze kontenera



Wnętrze kontenera



Wnętrze kontenera



chroot

To nie jest filesystem, którego szukasz

chroot

To nie jest filesystem, którego szukasz

```
# Run these commands before chrooting into the directory /mnt
mount -t proc proc /mnt/proc/
mount -t sysfs sys /mnt/sys/
mount -o bind /tmp /mnt/tmp/
mount -o bind /dev /mnt/dev/

chroot /mnt
```

namespace

Nie dotykaj moich procesów

namespace

Nie dotykaj moich procesów

```
unshare --pid --user --fork --mount-proc ./hello-world
unshare --net --user --fork ./hello-world
```

cgroup

Bo dobrze jest się dzielić

cgroup

Bo dobrze jest się dzielić

```
cgcreate -g cpu:test1
cgcreate -g memory:test2

cgset -r cpu.cfs_quota_us=50000 test1
cgset -r memory.limit_in_bytes=9999999 test2

cgexec -g cpu:test1 ./hello-world
cgexec -g memory:test2 ./hello-world
```

runc

Dokładnie jak docker

runc

Dokładnie jak docker

```
runc spec
# edit config.json
# provide rootfs/

sudo runc create container-name
sudo runc start container-name
# sudo runc run container-name
```

To po co nam obrazy?

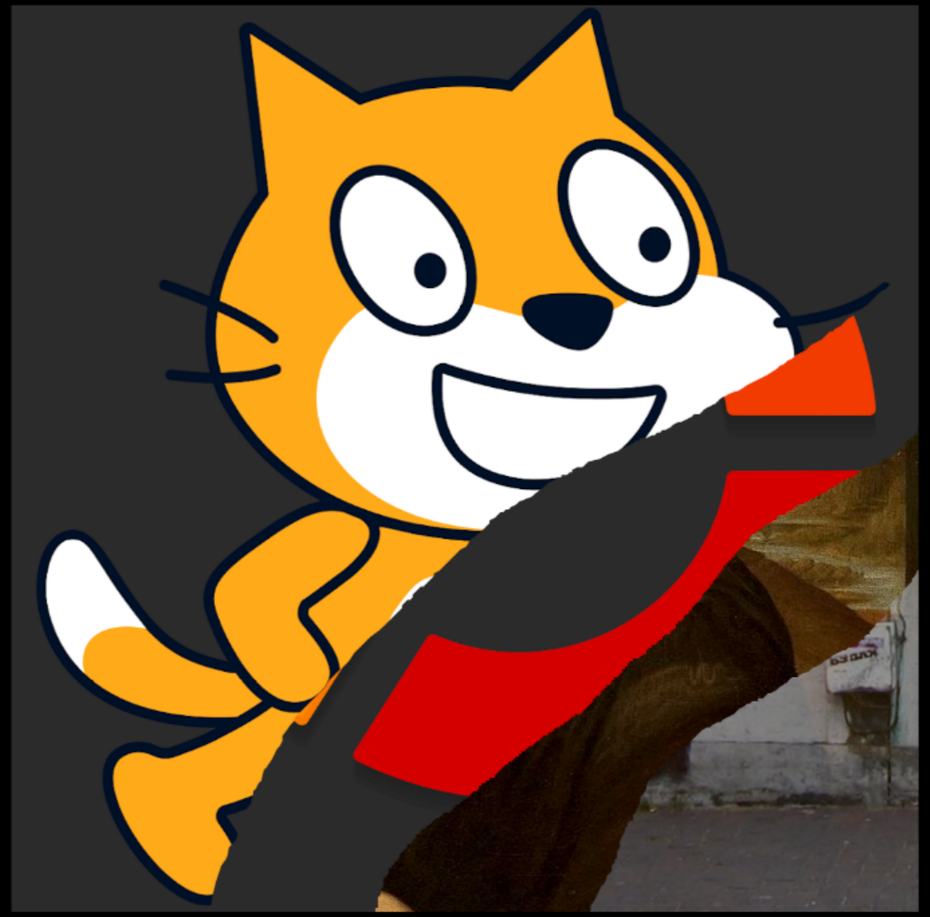
Obrazy mają warstwy

Obrazy... aż do końca

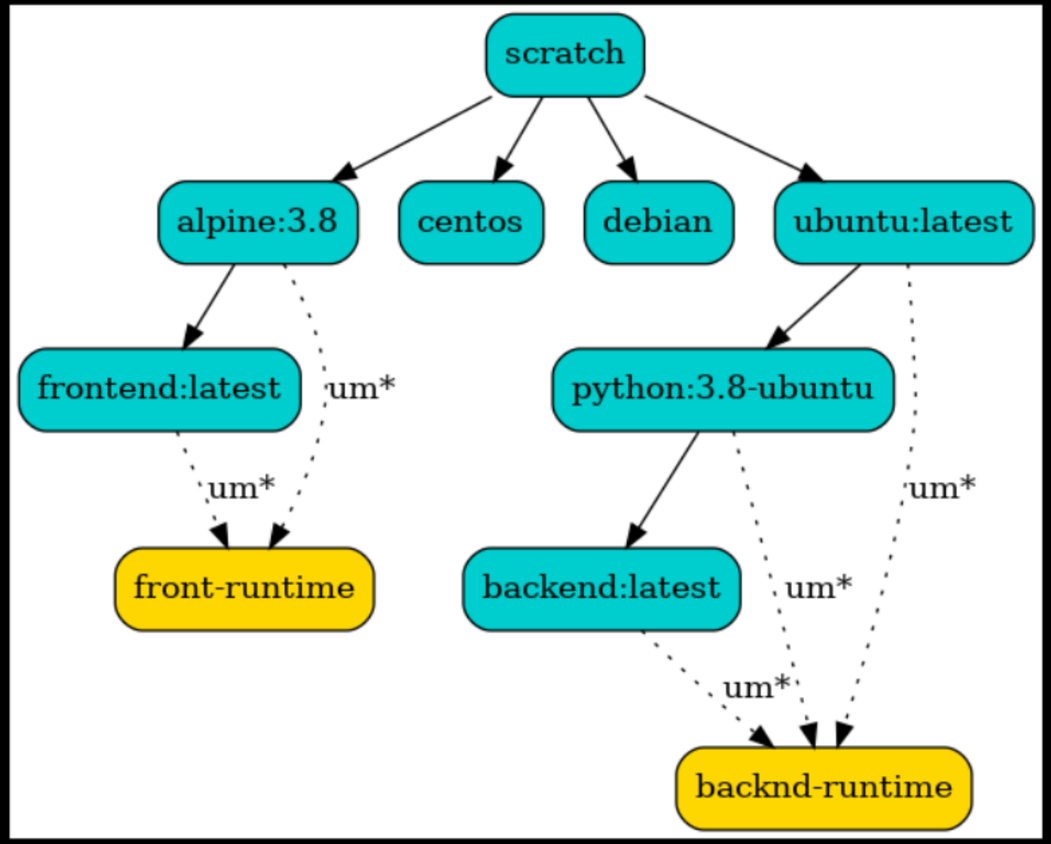
Obrazy... aż do końca

Obrazy... aż do końca

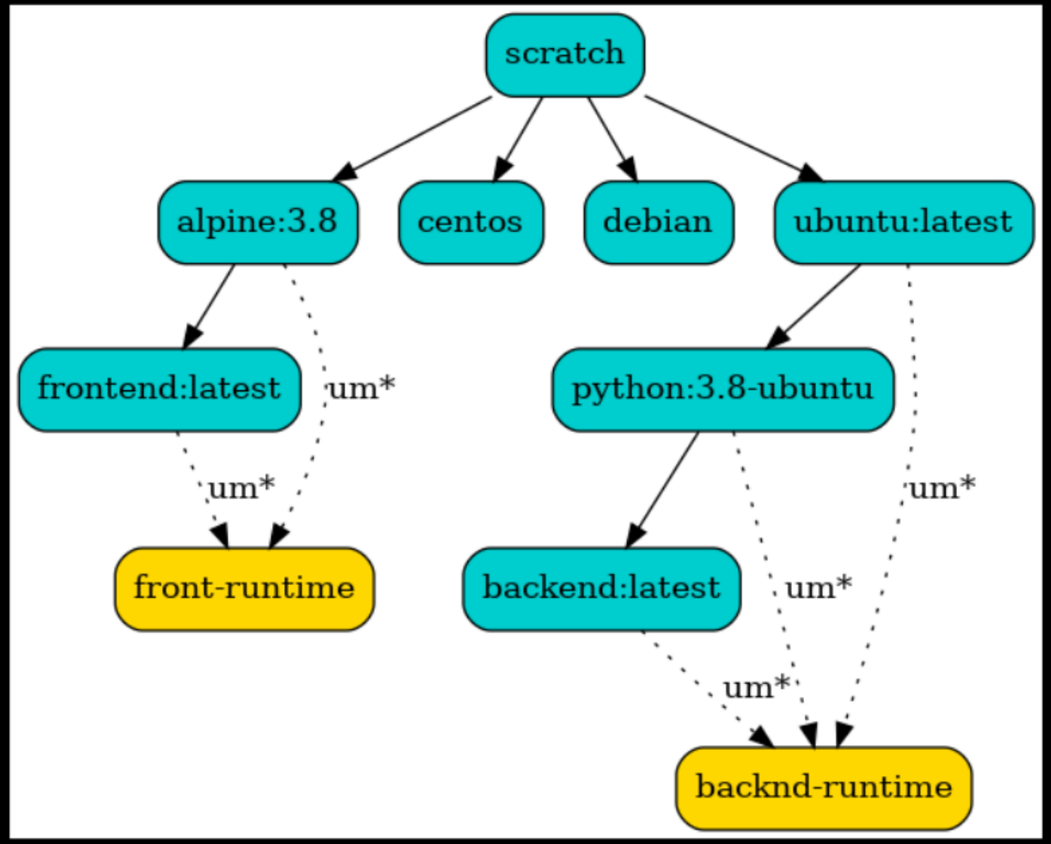
Obrazy... aż do końca



Dobrze znane warstwy

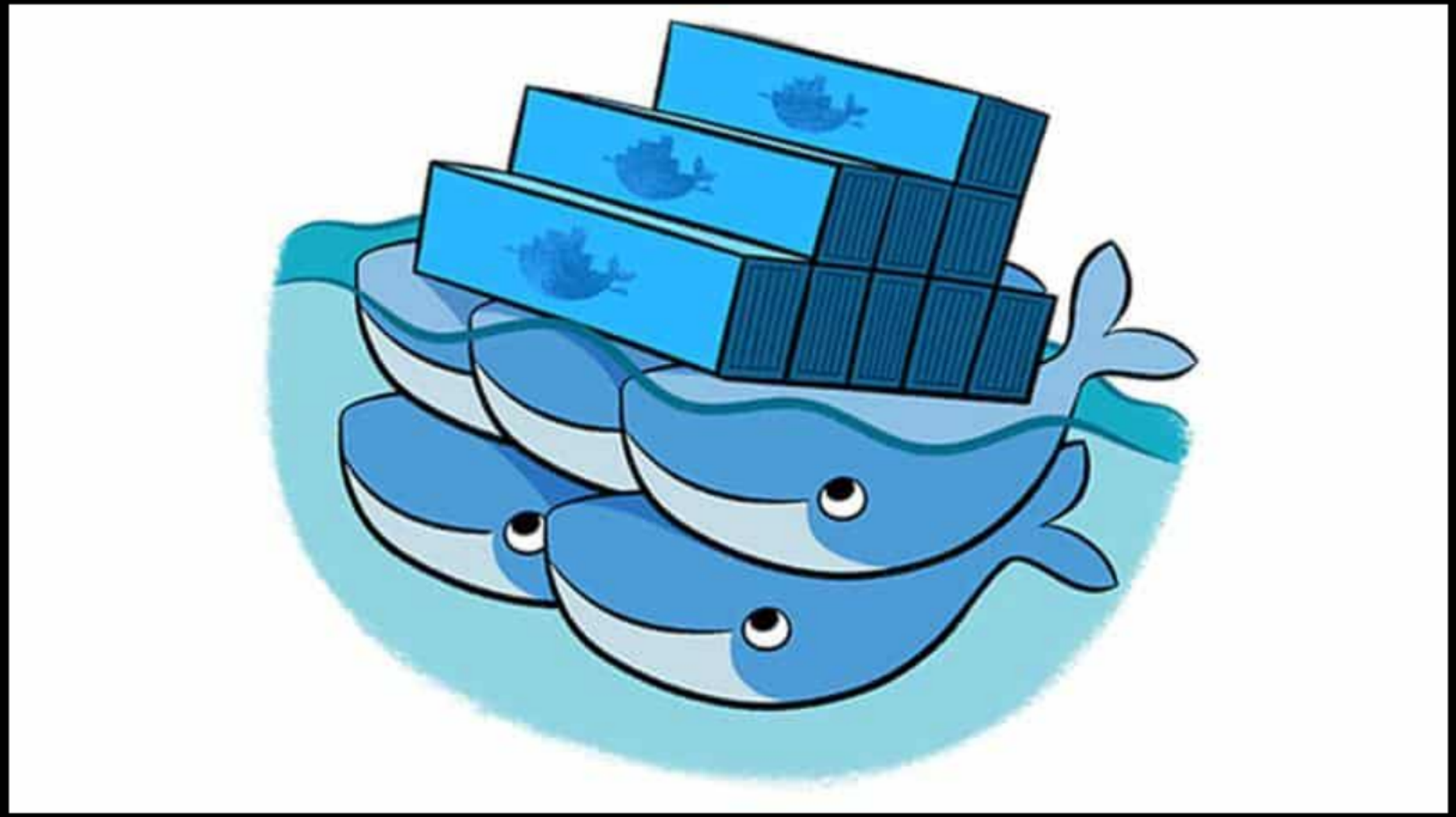


Dobrze znane warstwy



um - Union Mount

Trzymane na cudzych komputerach



TL;DR

- każdy może napisać `"dockera"` w skończonym czasie
- napisanie dockera™ już nie jest takie łatwe
- docker nie jest niezbędny, mamy dostępne alternatywy
- nadal warto dockera poznać w pierwszej kolejności, skoro `"docker jest wszędzie"`
- spektrum wirtualizacji jest trochę jak NoSQL - wielowymiarowe

PGS SOFTWARE
Part of Xebia

https://norasoft.eu/talks/container-magic/

mborkowski@pgs-soft.com