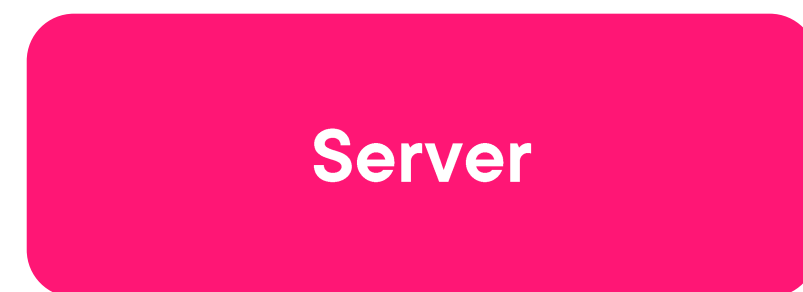


**Up Next:**

# **Docker Architecture: Big Picture**

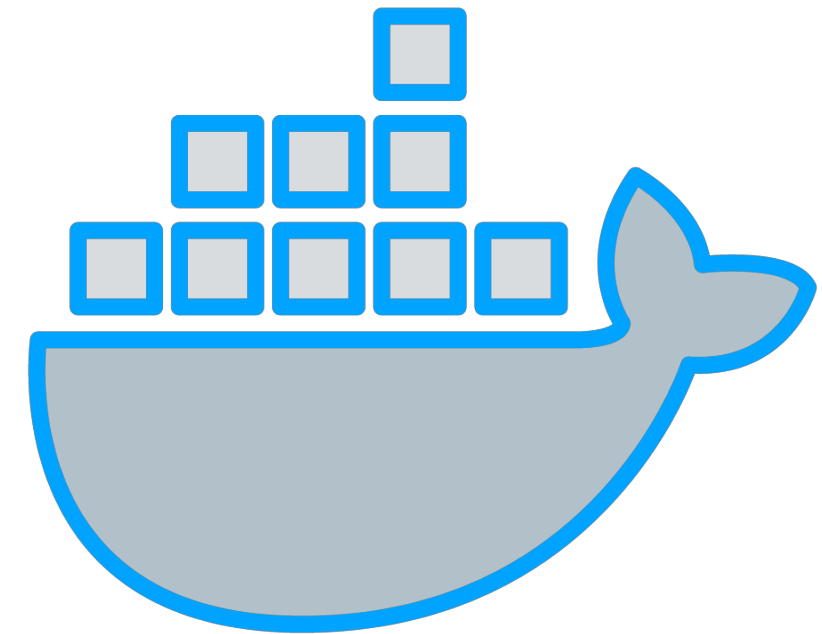
---





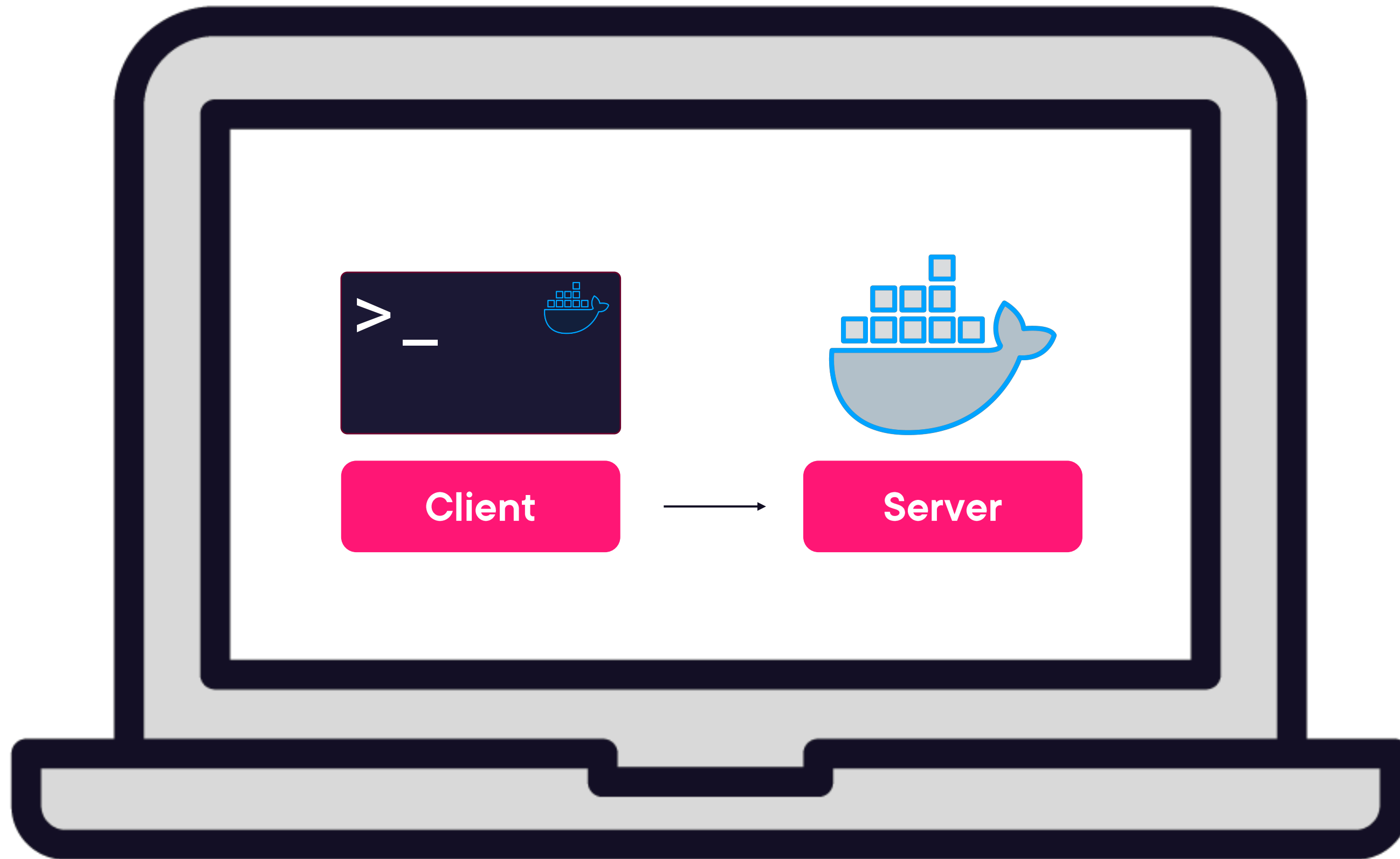


Client



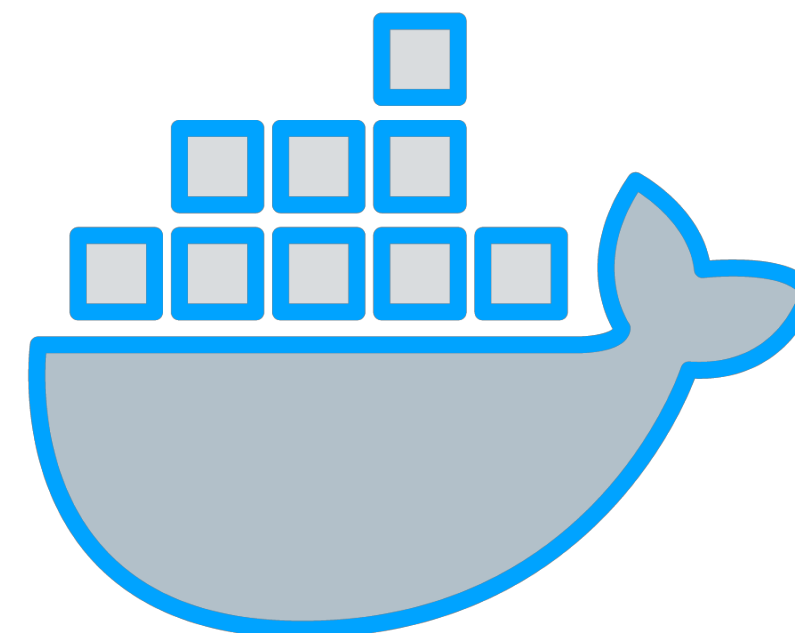
Server



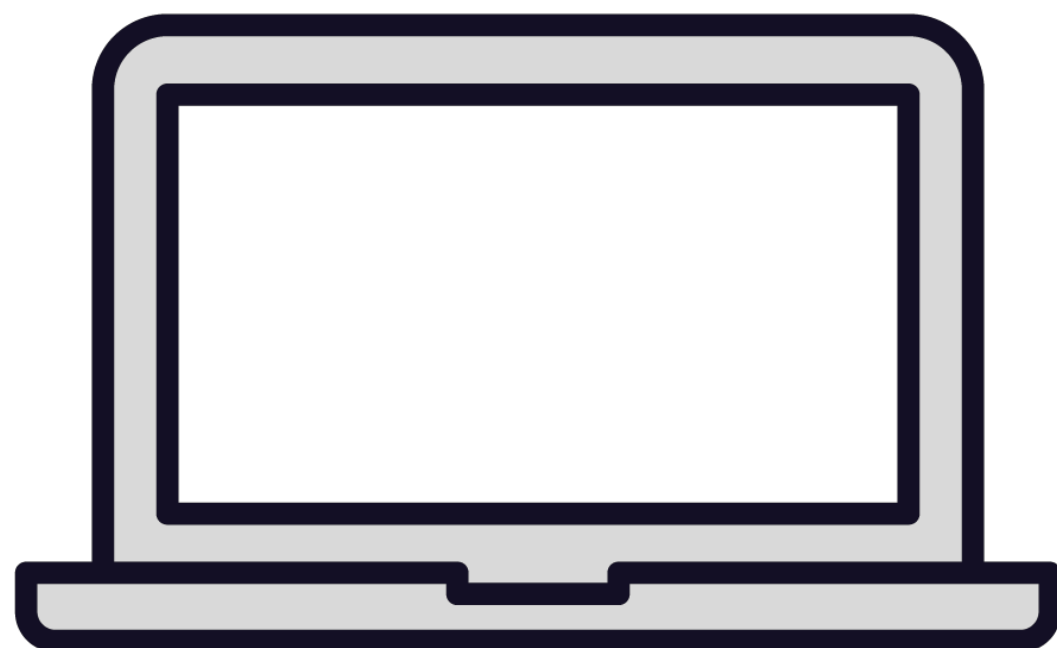




Client



Server



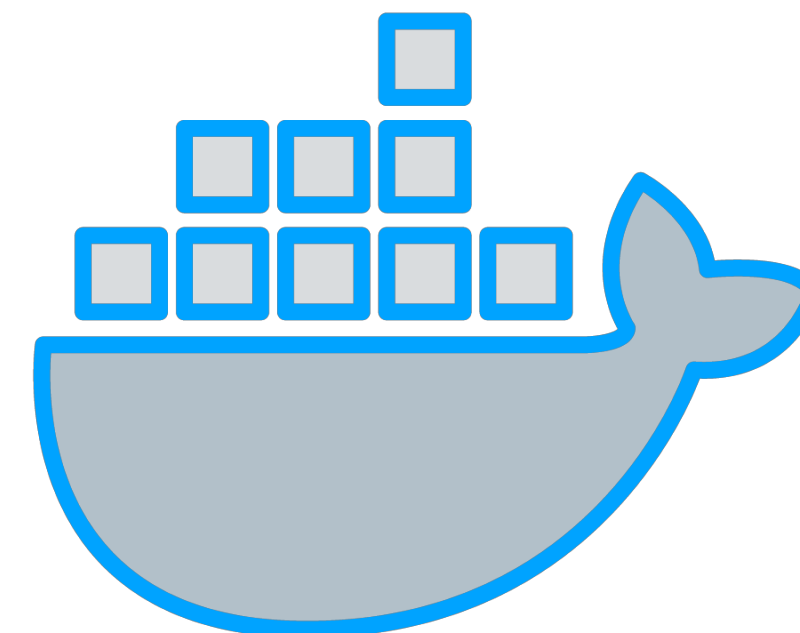


Client

POST

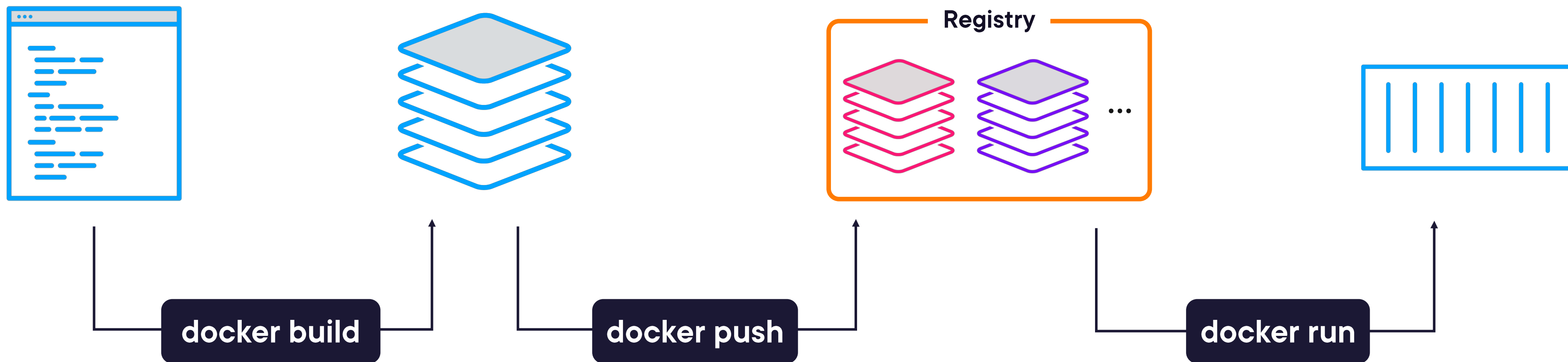


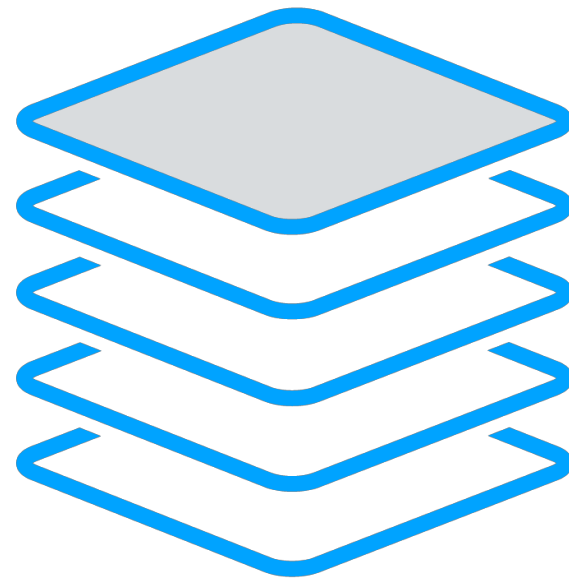
{API}



Server

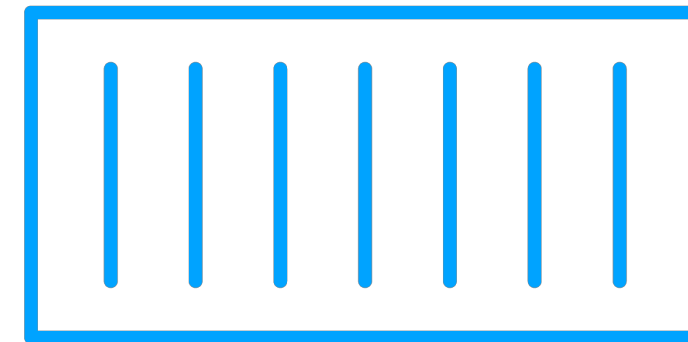






**docker build**

OCI image



**docker run**

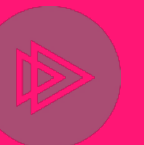
OCI container

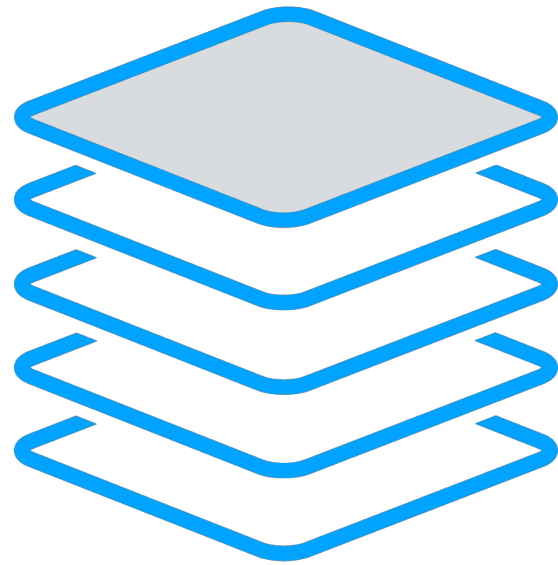




# Open Container Initiative (OCI)

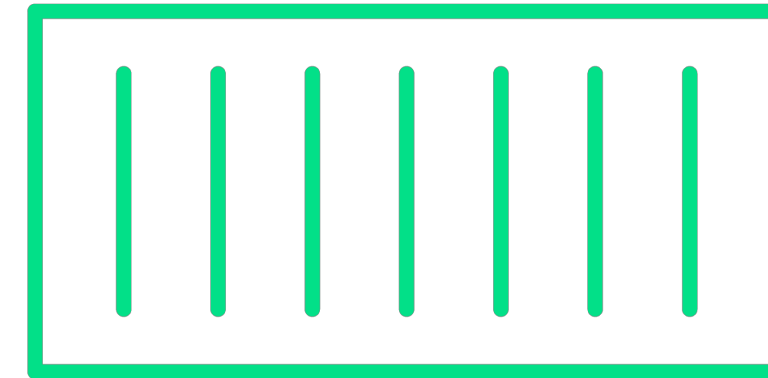
Standards body for low-level container-related standards.





**All mean the same**

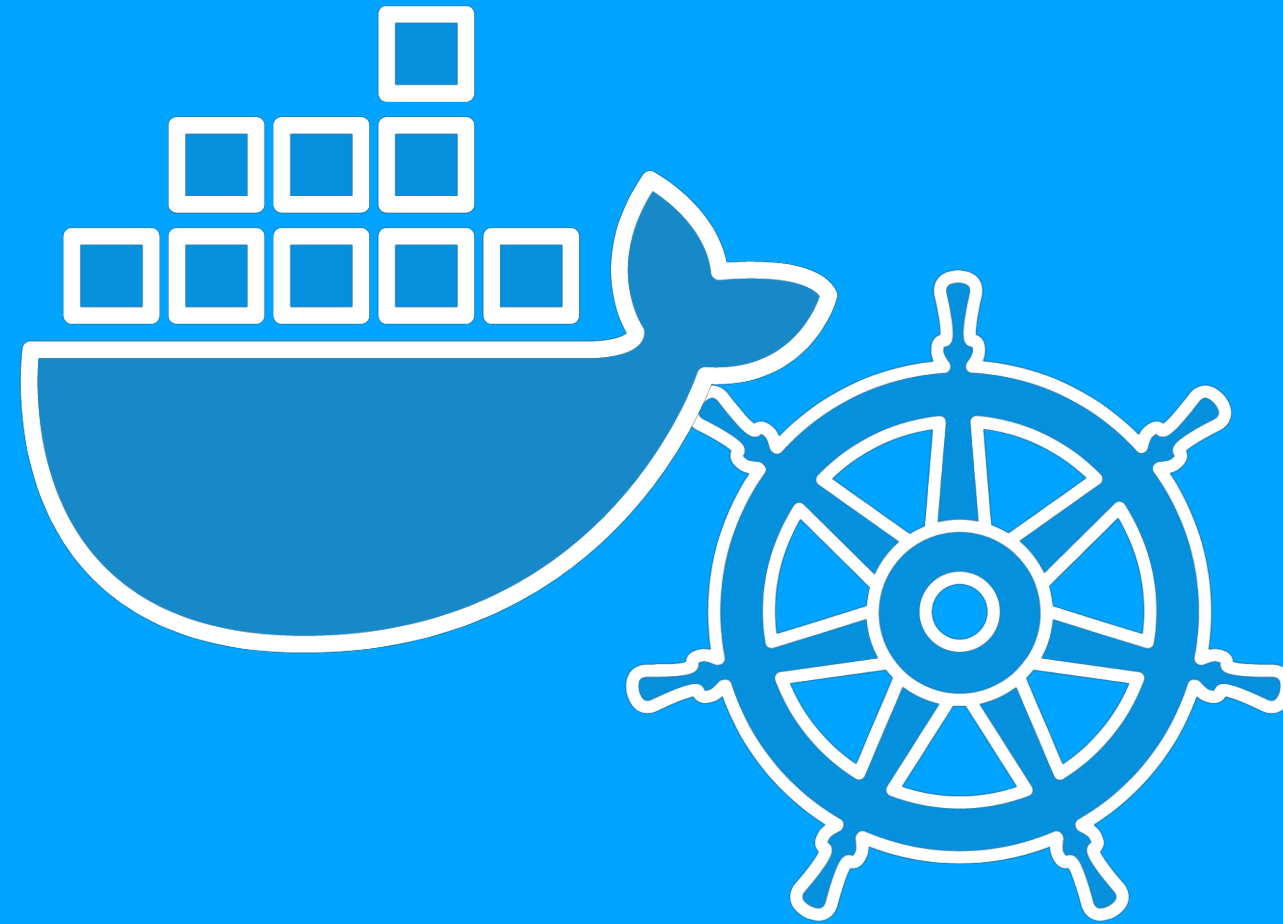
**Image**  
**Container image**  
**Docker Image**  
**OCI image**



**All mean the same**

**Container**  
**Docker container**  
**OCI container**





# Docker and Kubernetes

Both use and work with OCI images and containers...



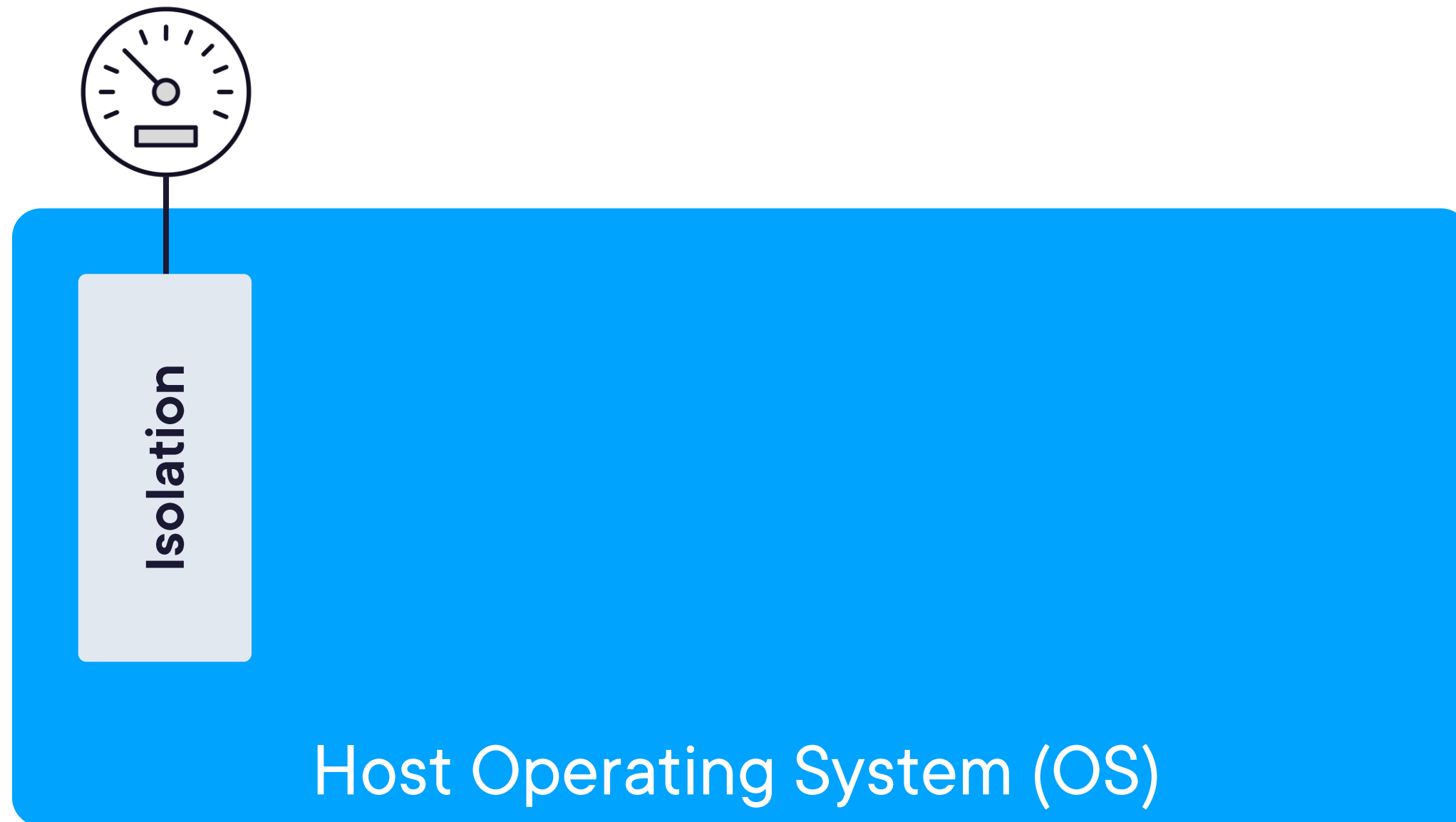
# Container

Isolated area of OS with resource limits applied.



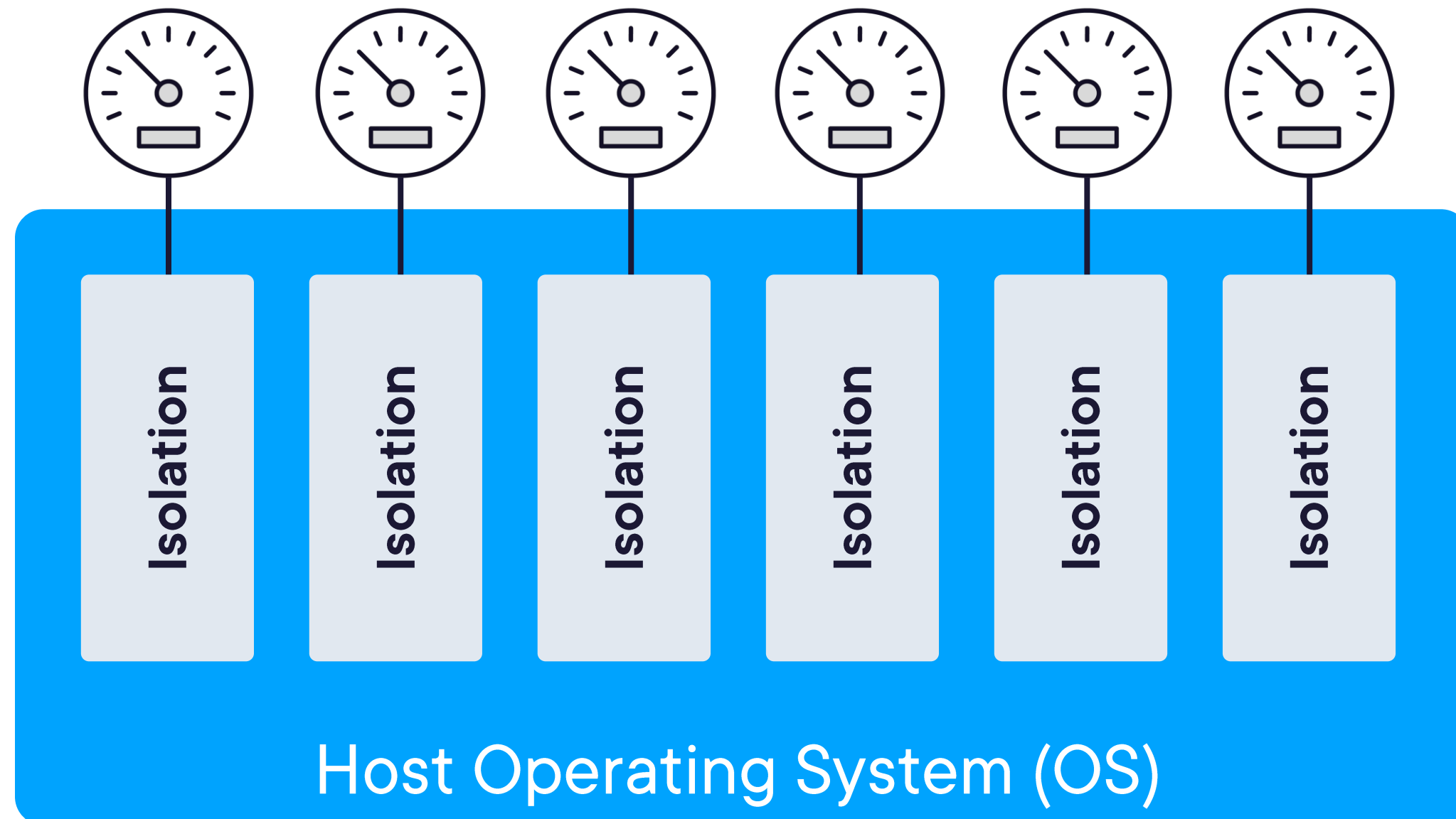
# Container

Isolated area of OS with resource limits applied.



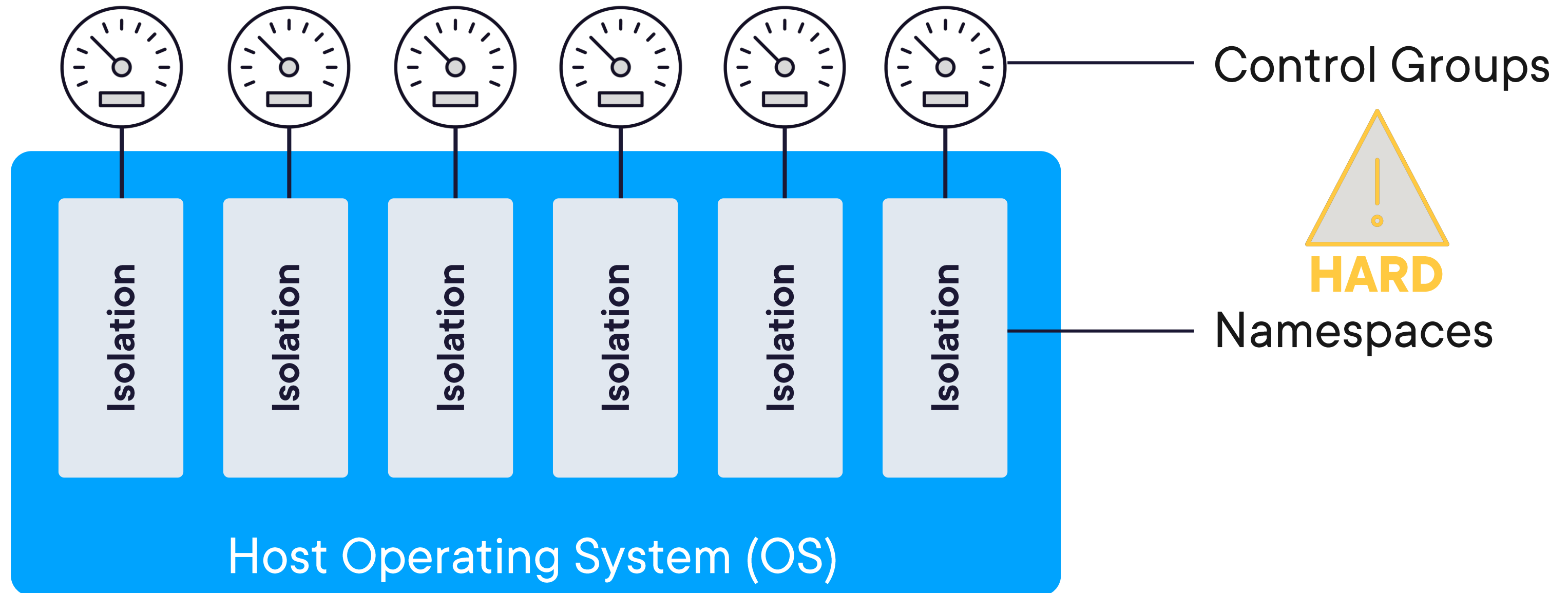
# Container

Isolated area of OS with resource limits applied.



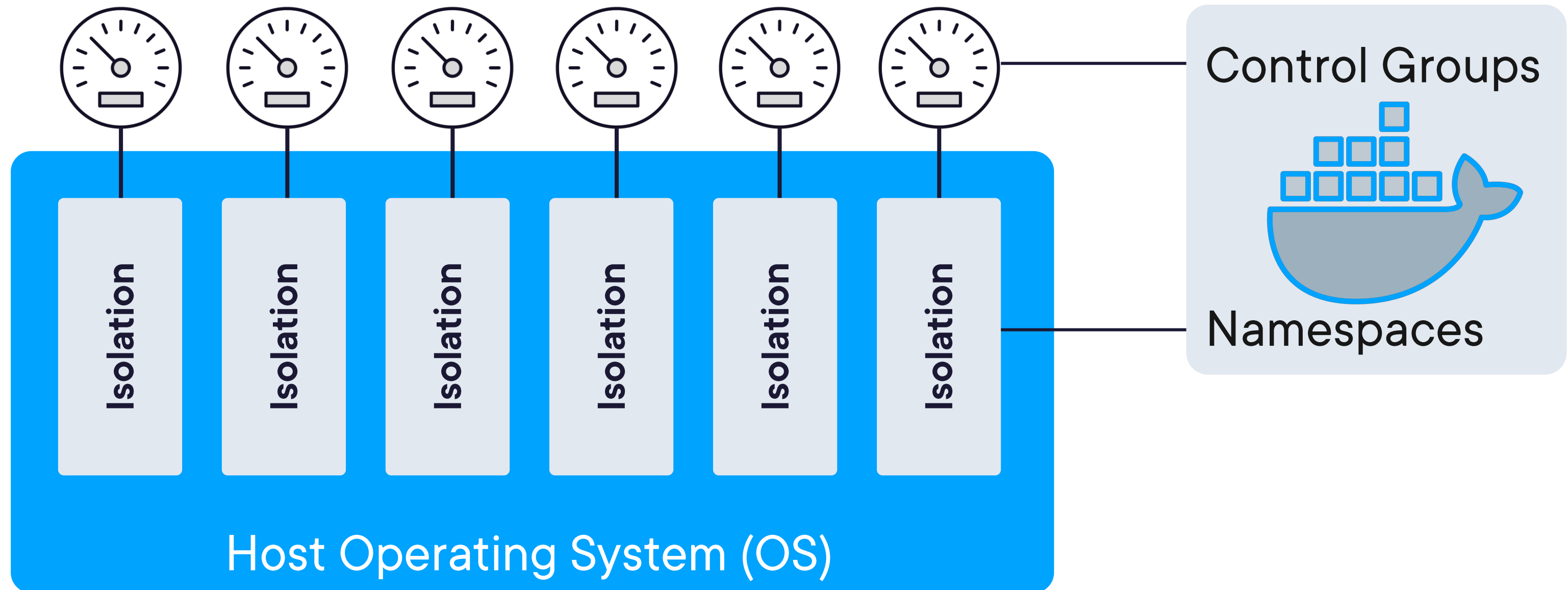
# Container

Isolated area of OS with resource limits applied.

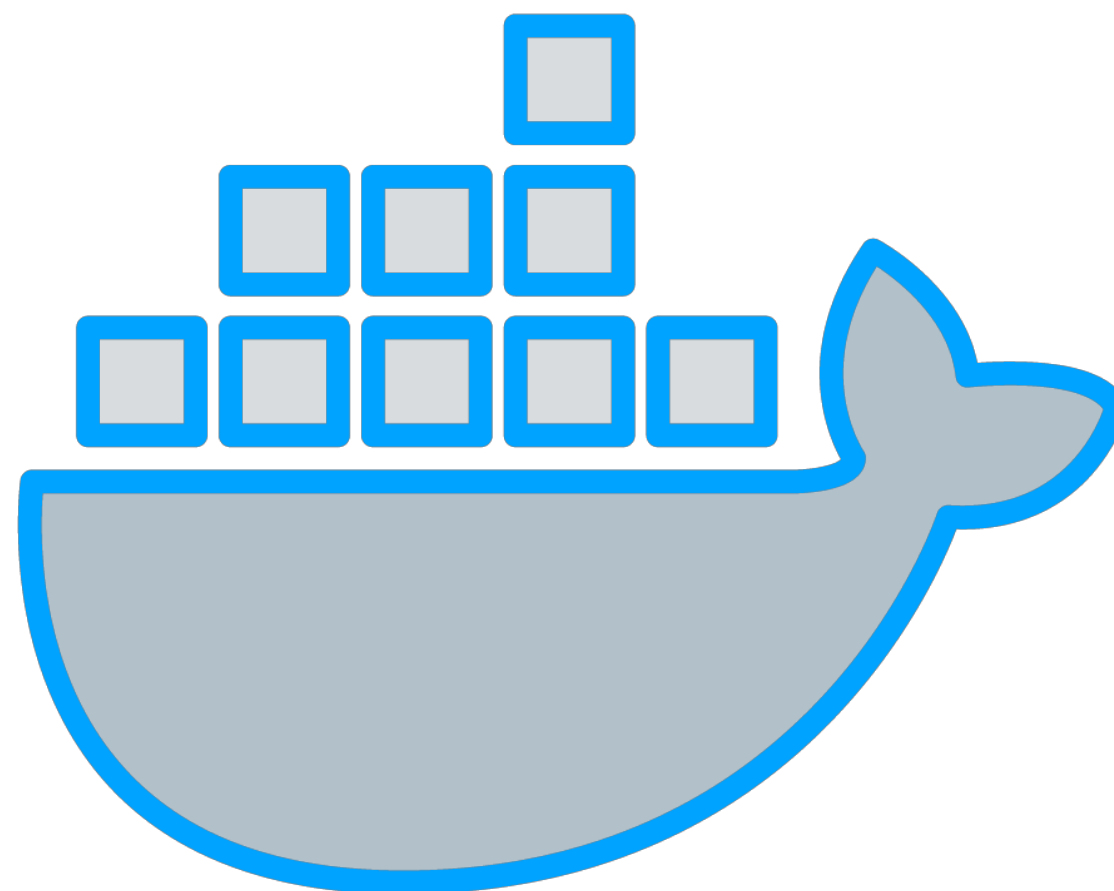


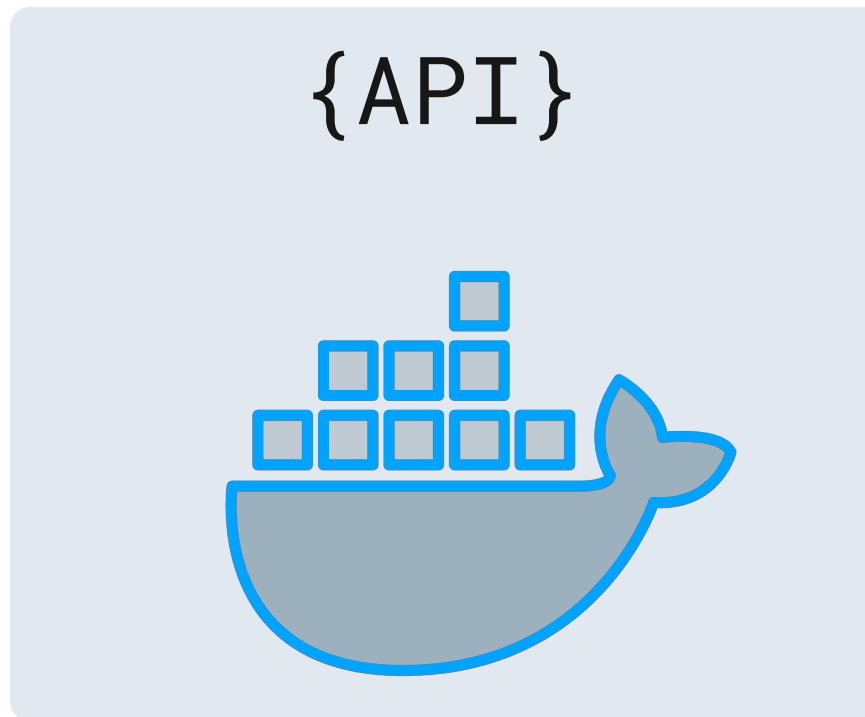
# Container

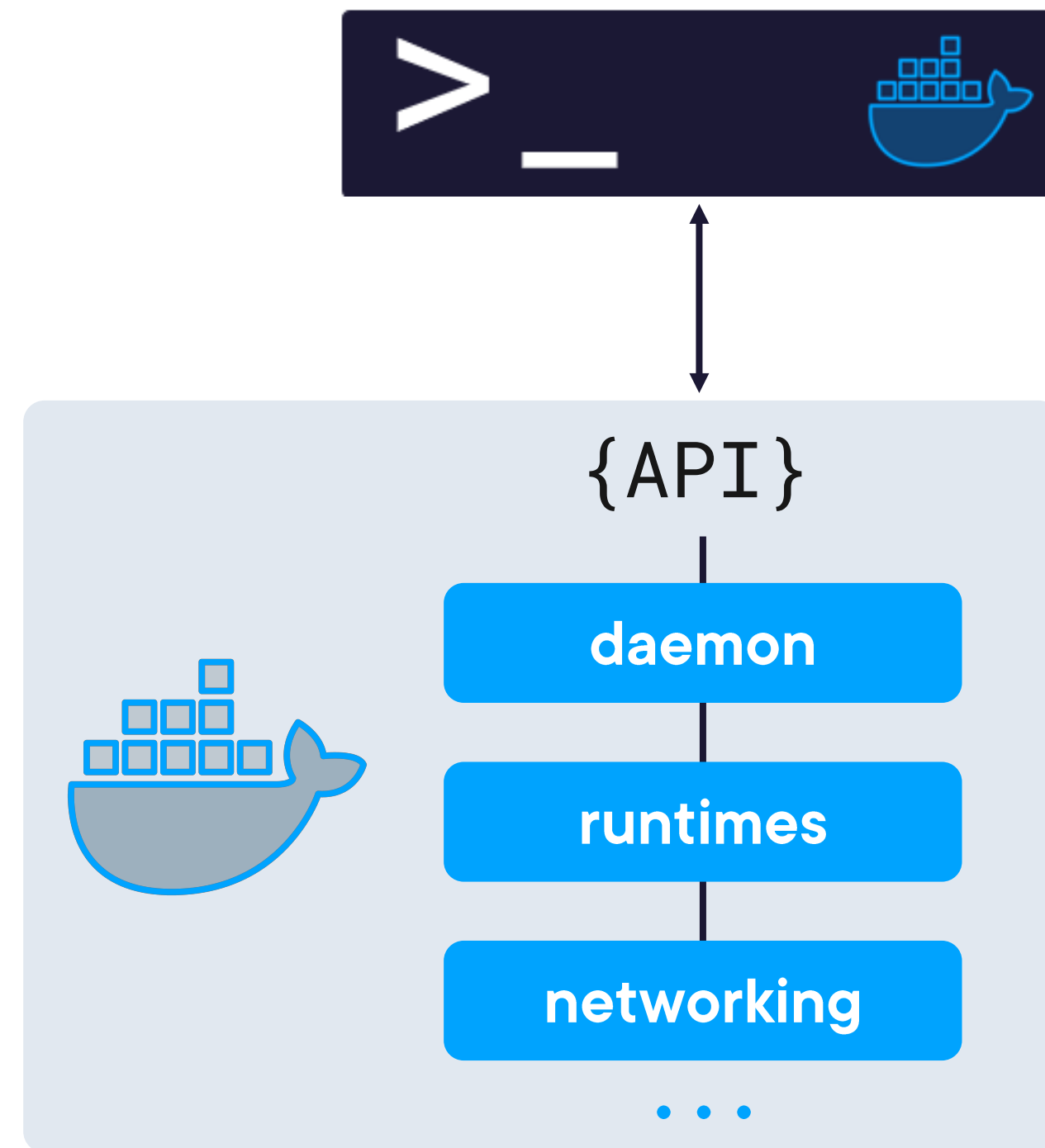
Isolated area of OS with resource limits applied.

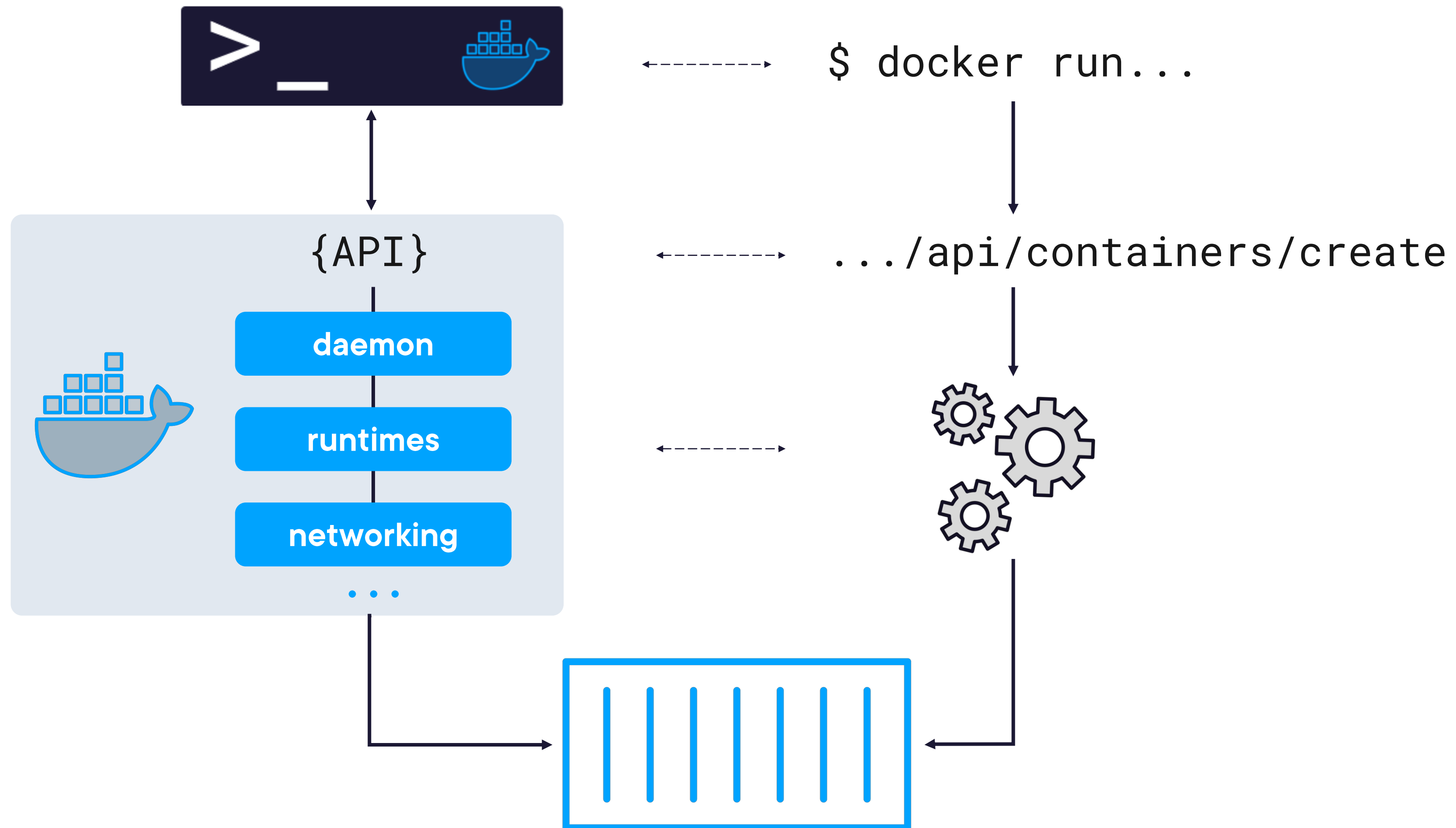










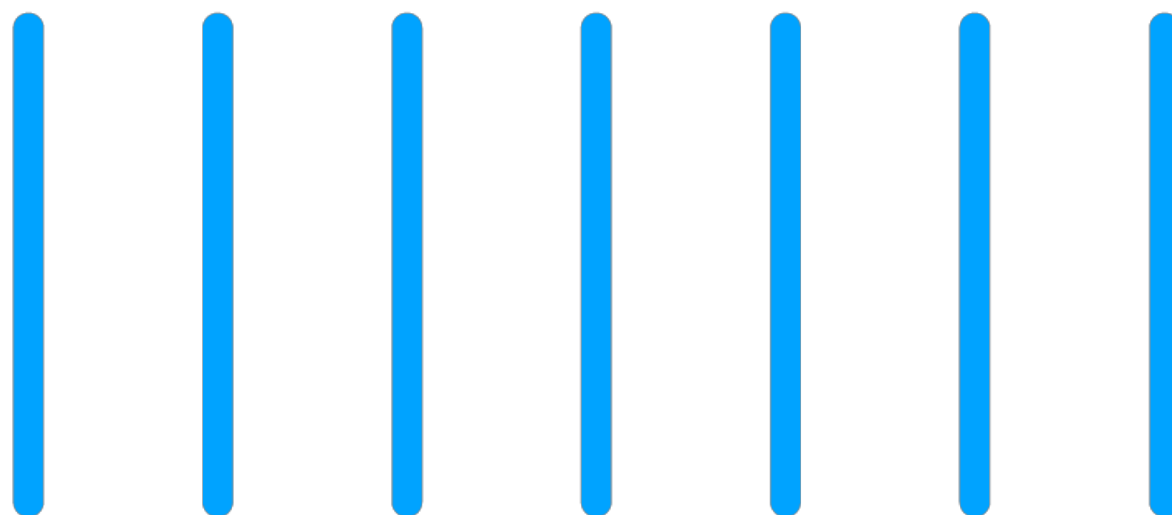


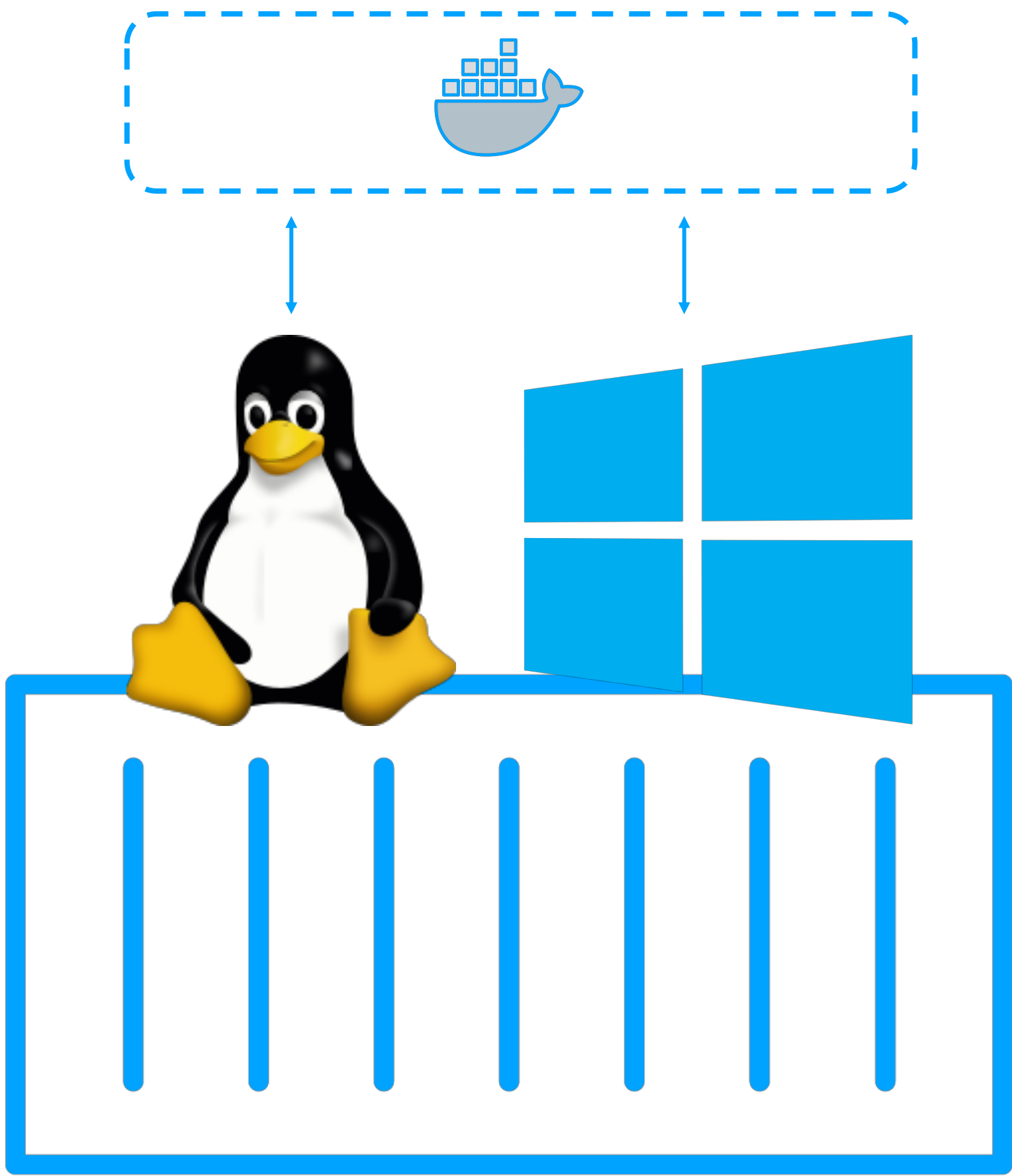
**Up Next:**

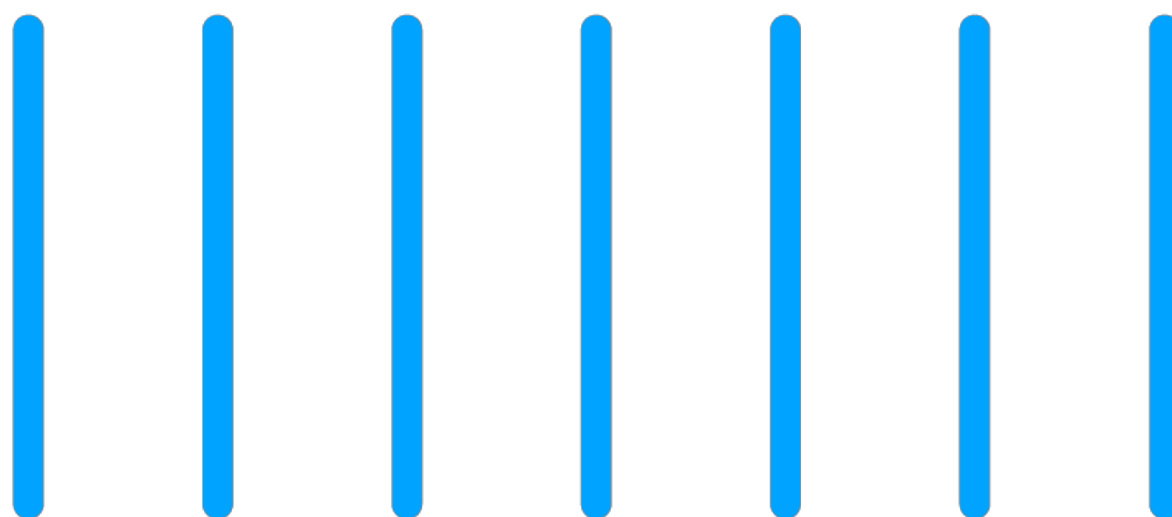
# **Kernel Internals**

---











# Fundamental Container Primitives in Kernel

## Namespaces

All about isolation

## Control Groups

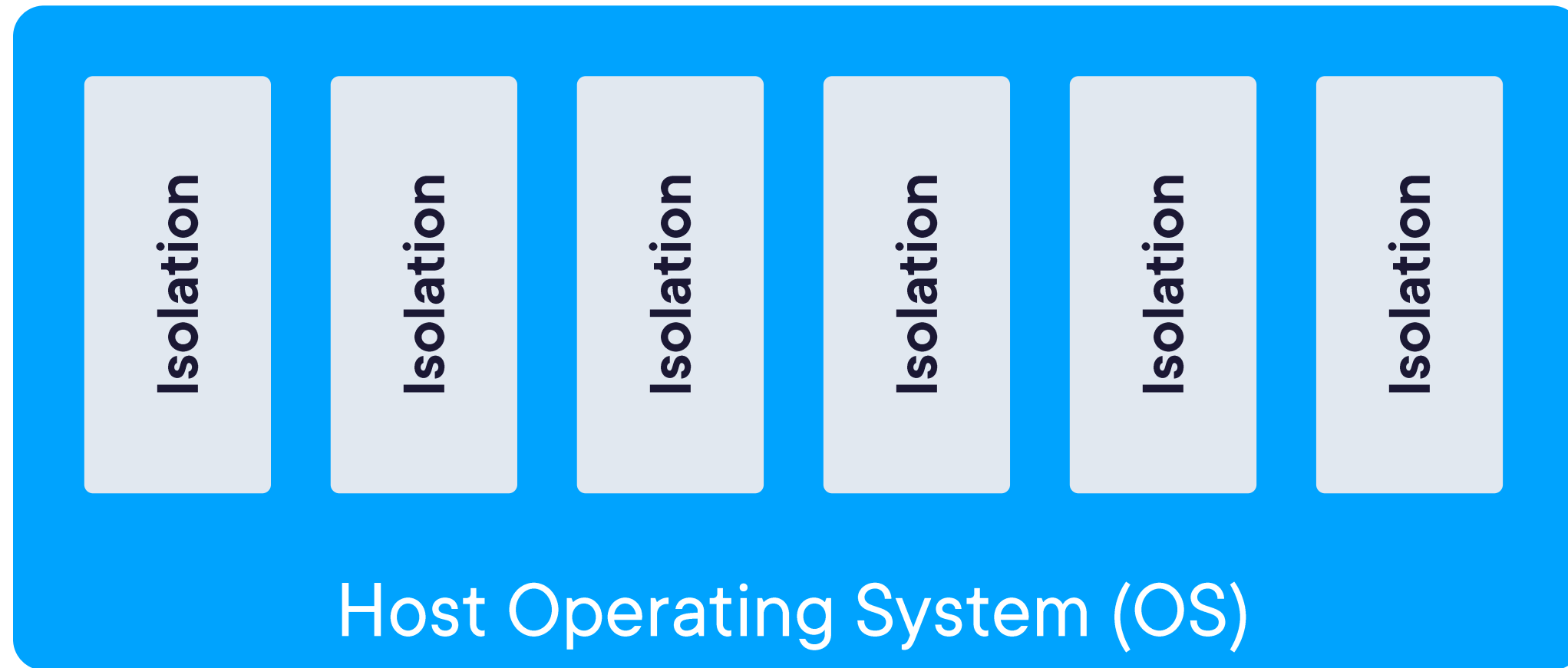
All about limits



**Namespaces**  
All about isolation

**Control Groups**  
All about limits

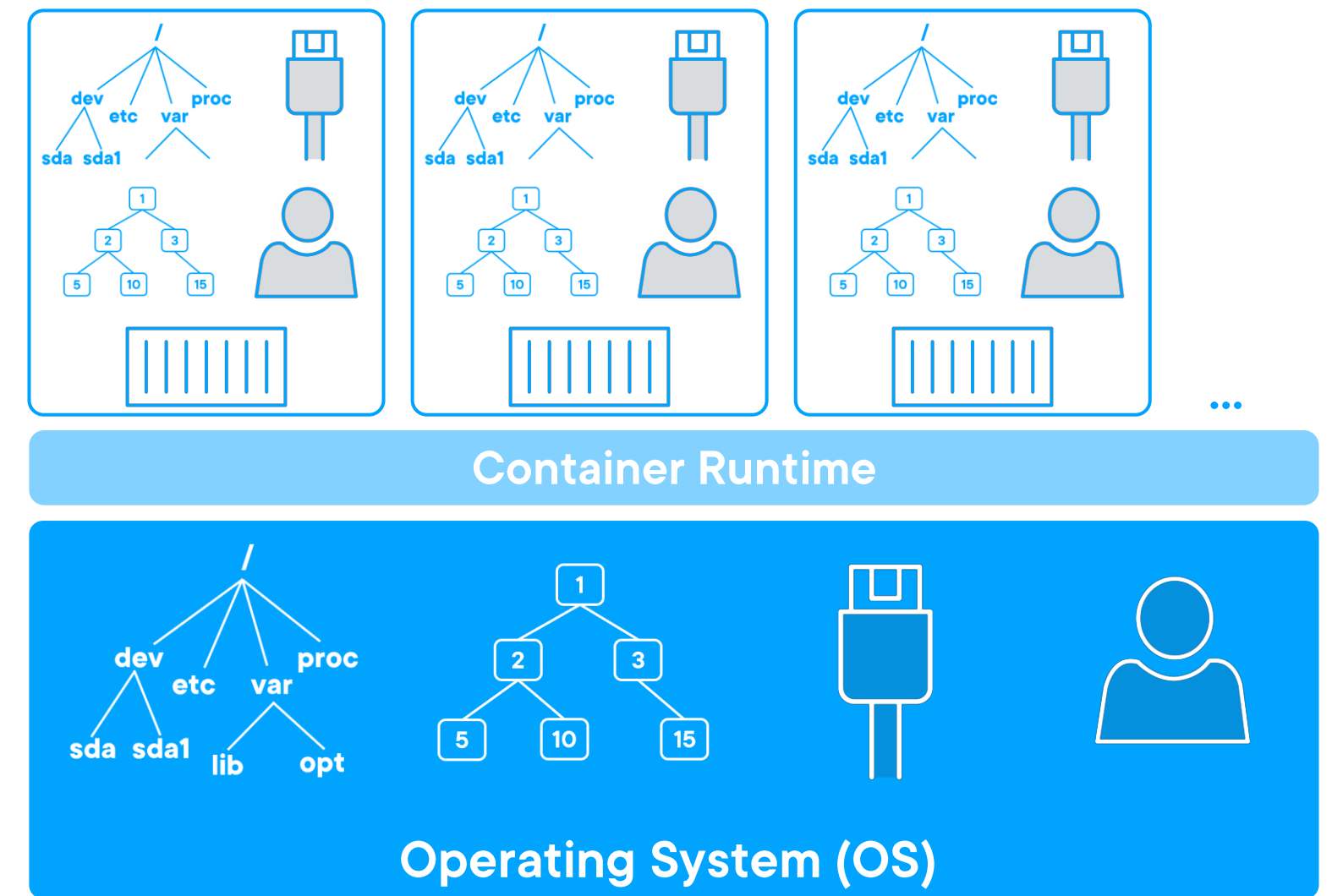
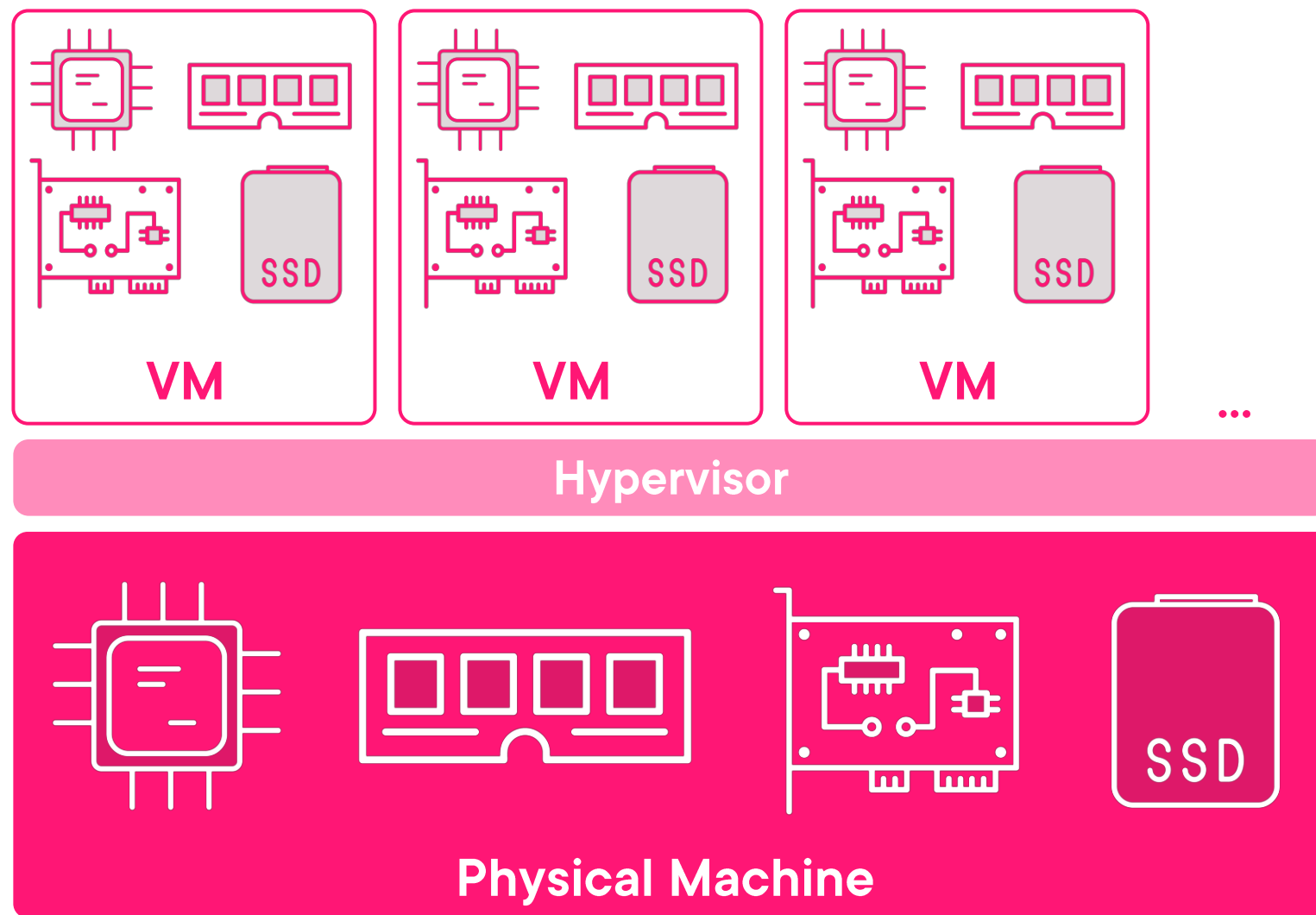




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

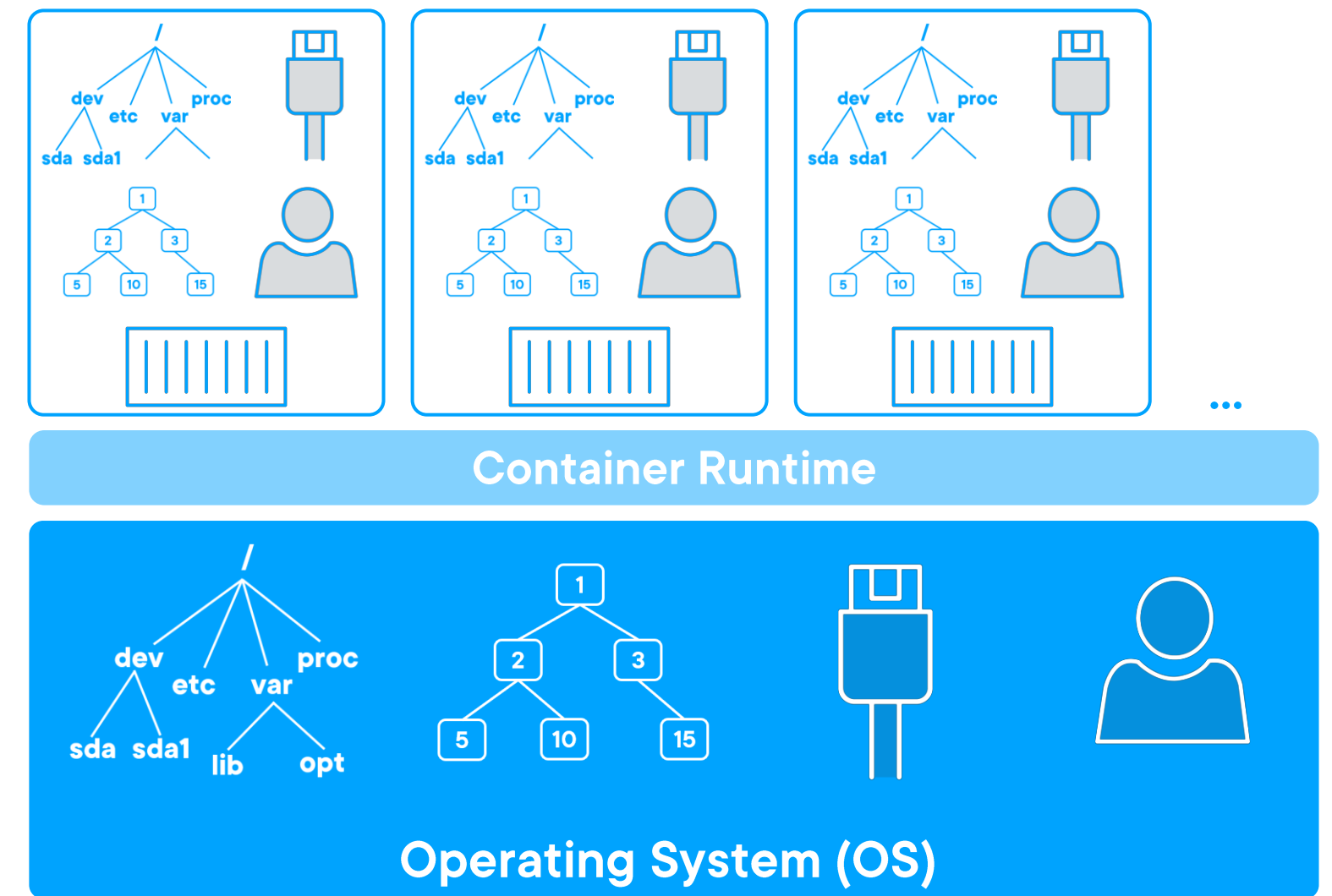
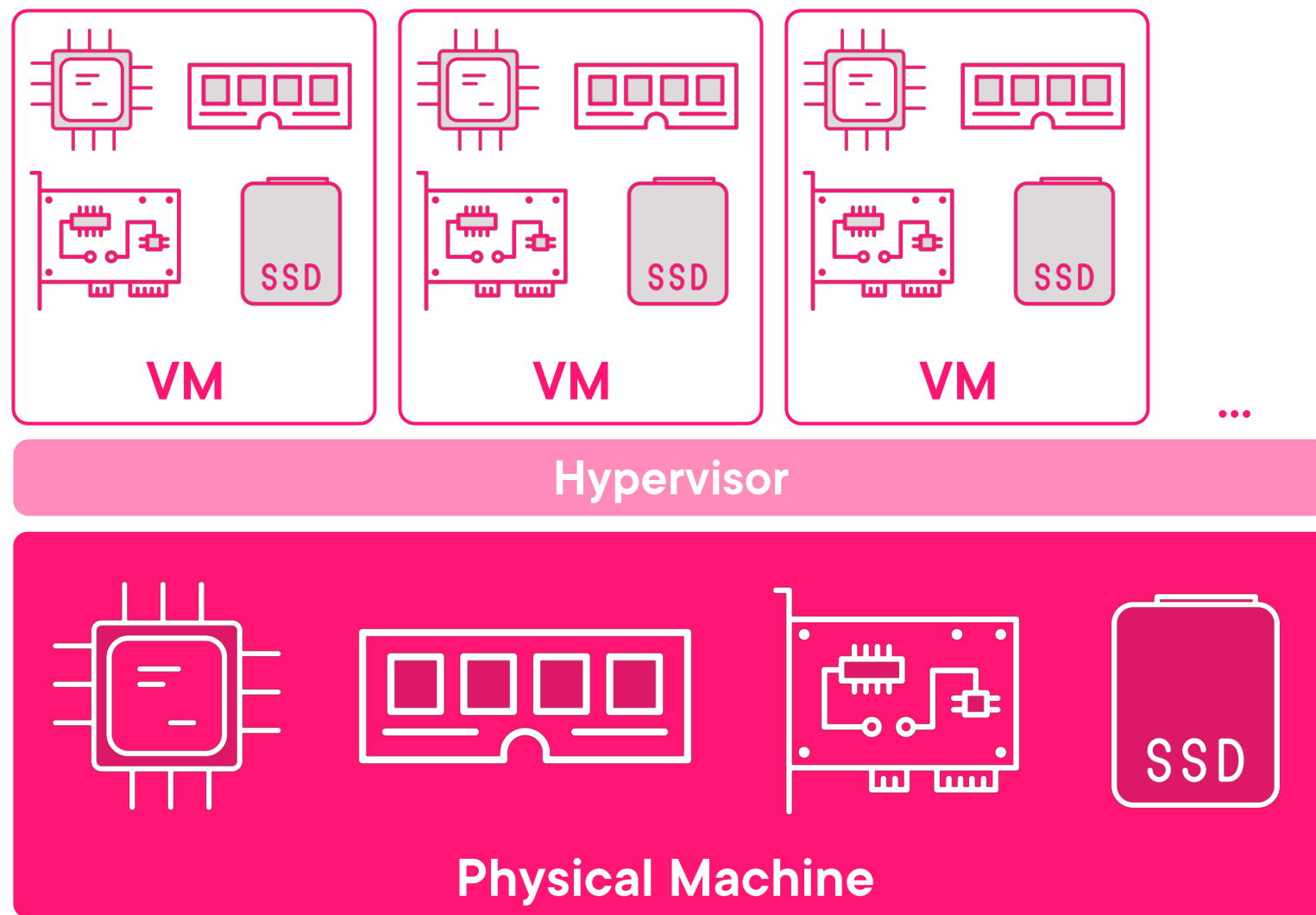




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

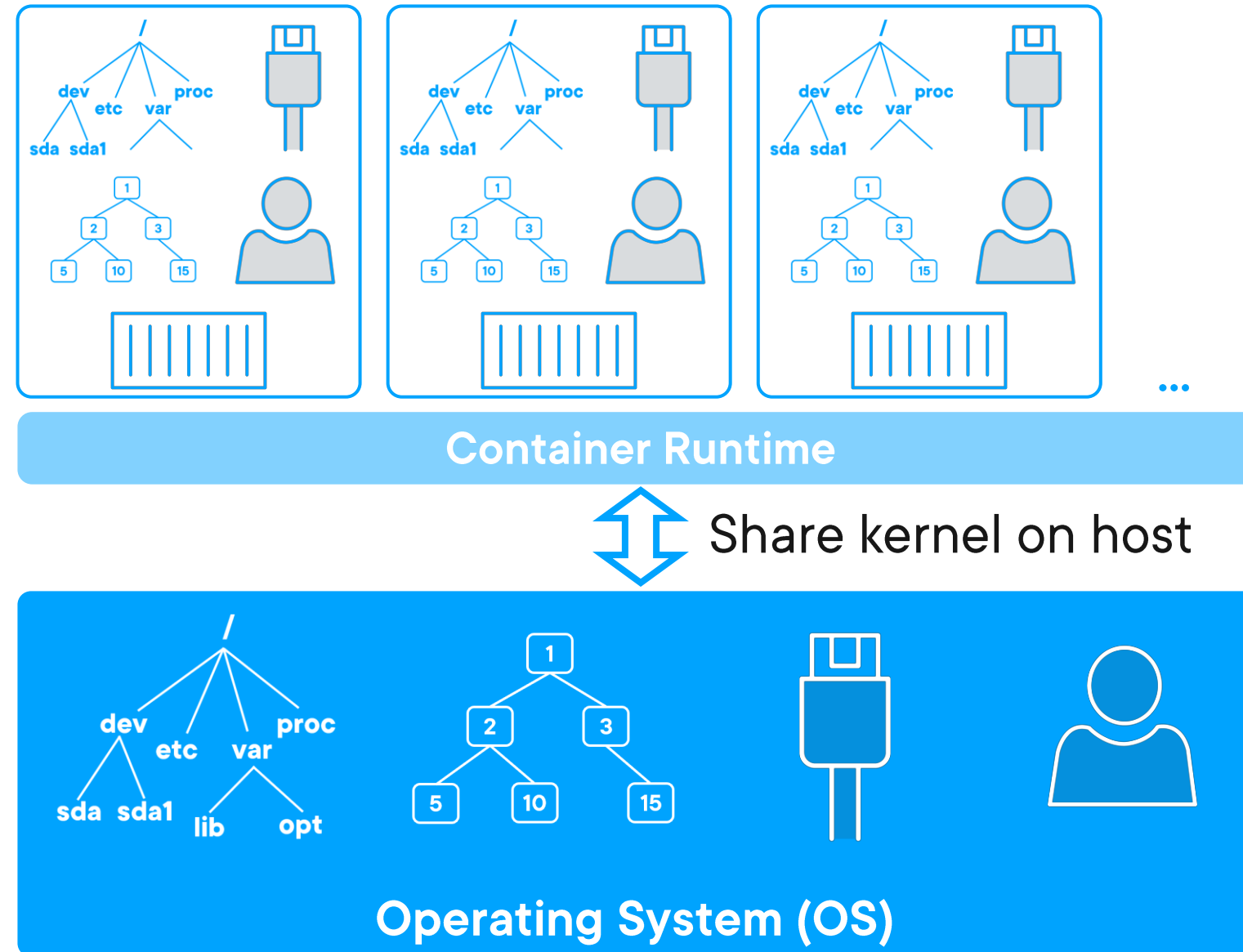




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

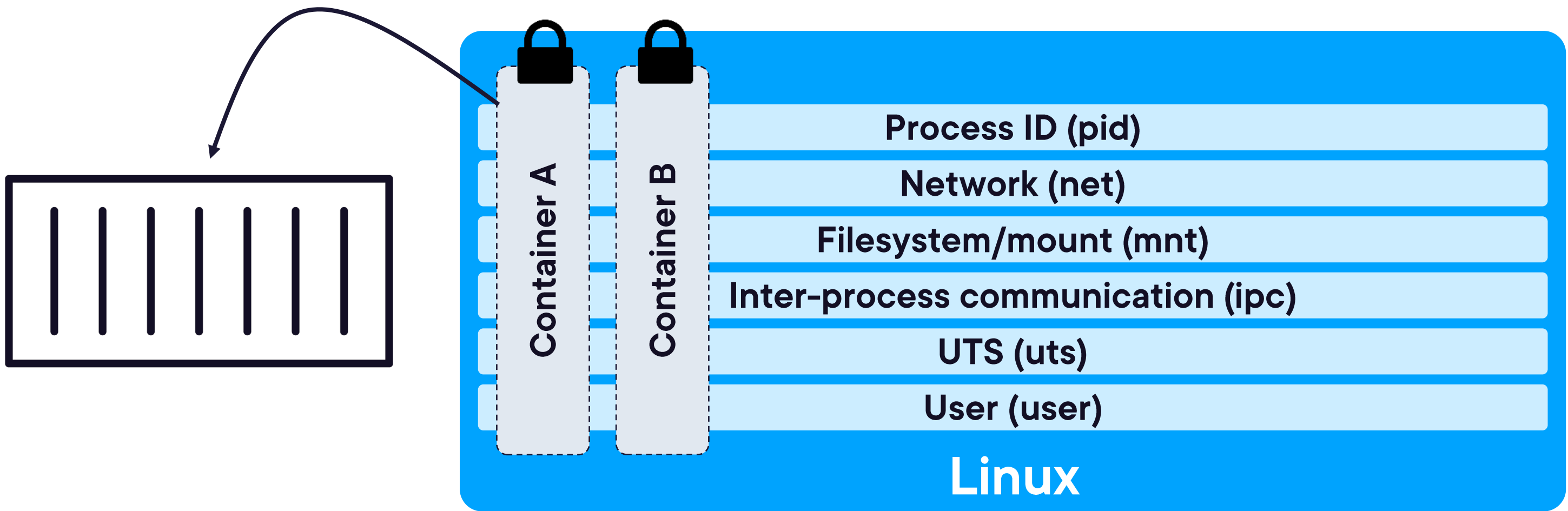




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

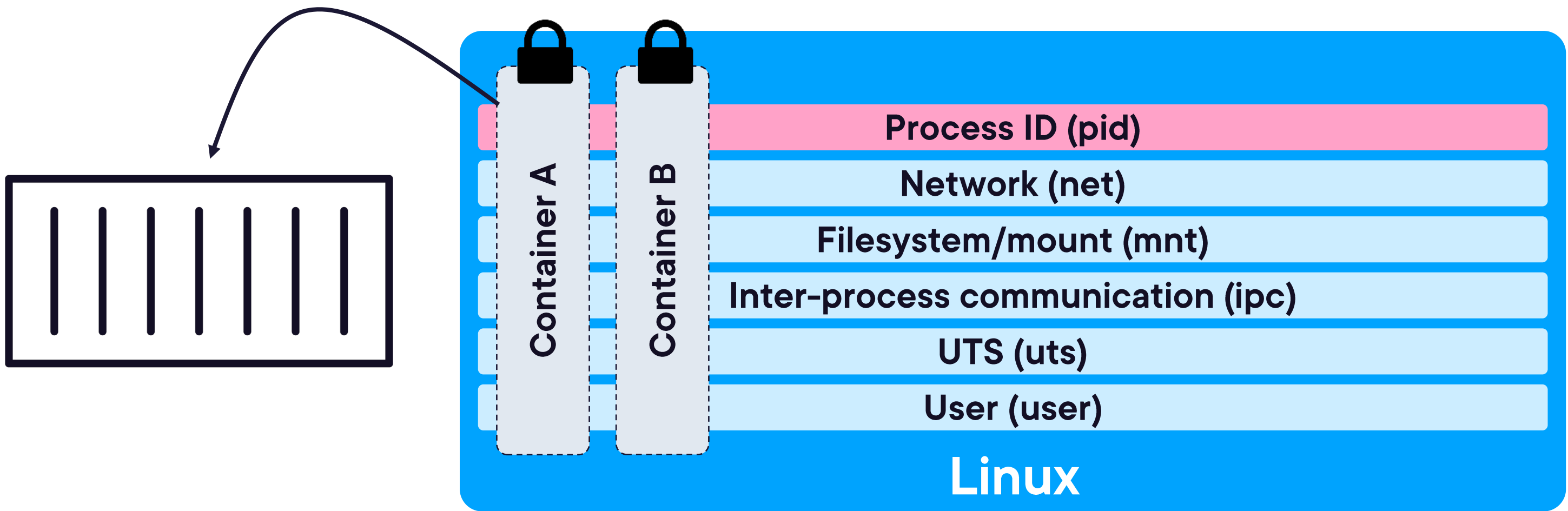




**Namespaces**  
All about isolation

**Control Groups**  
All about limits



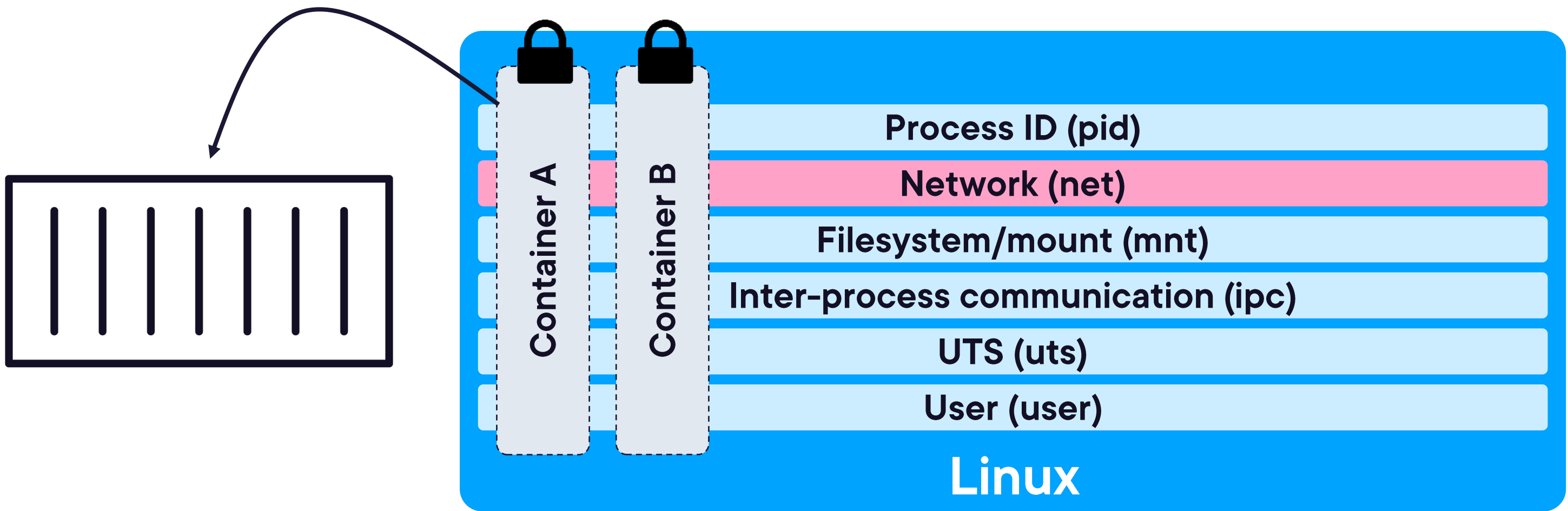


**Namespaces**  
All about isolation

**Control Groups**  
All about limits



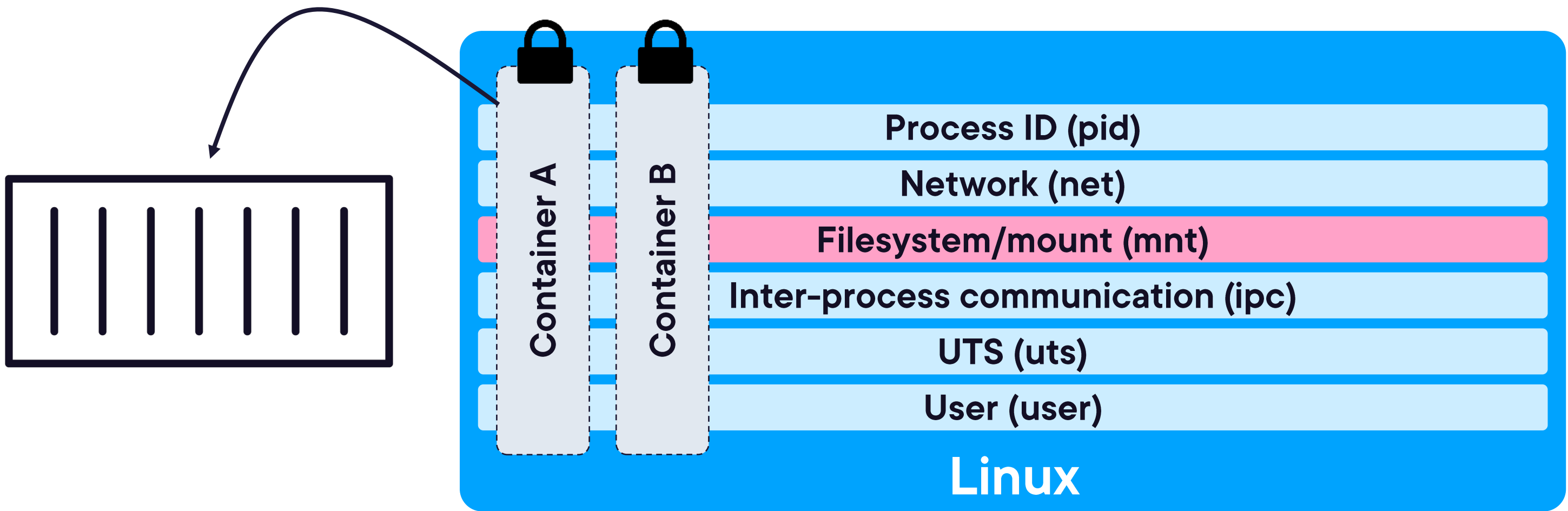




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

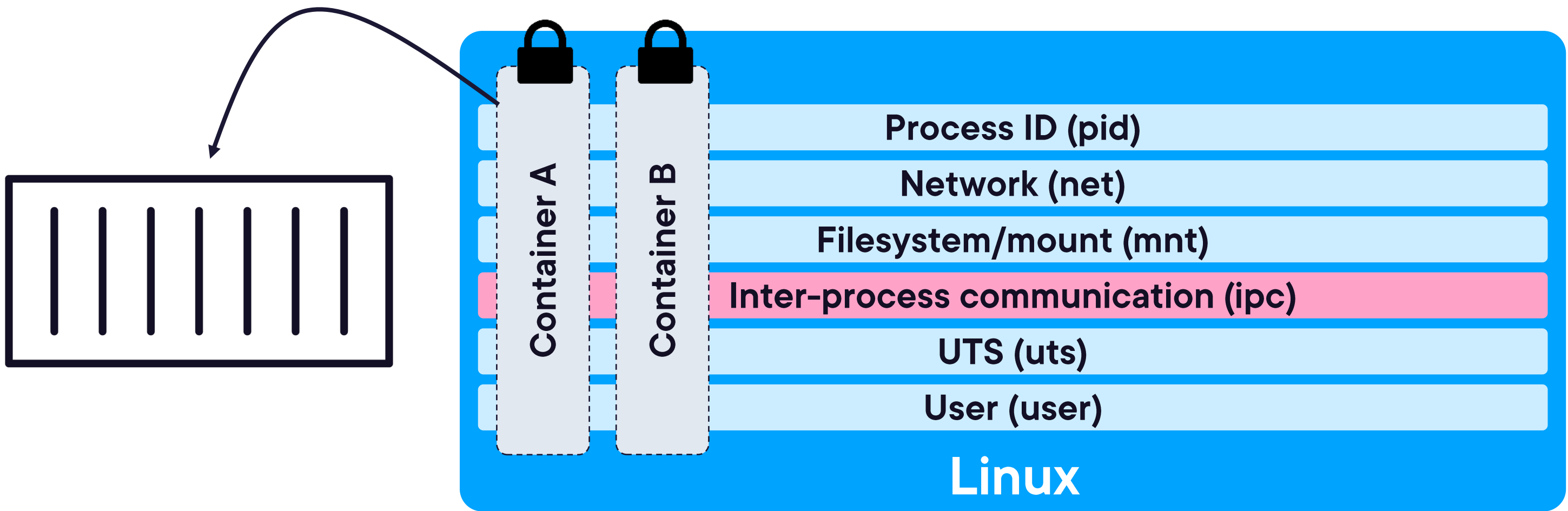




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

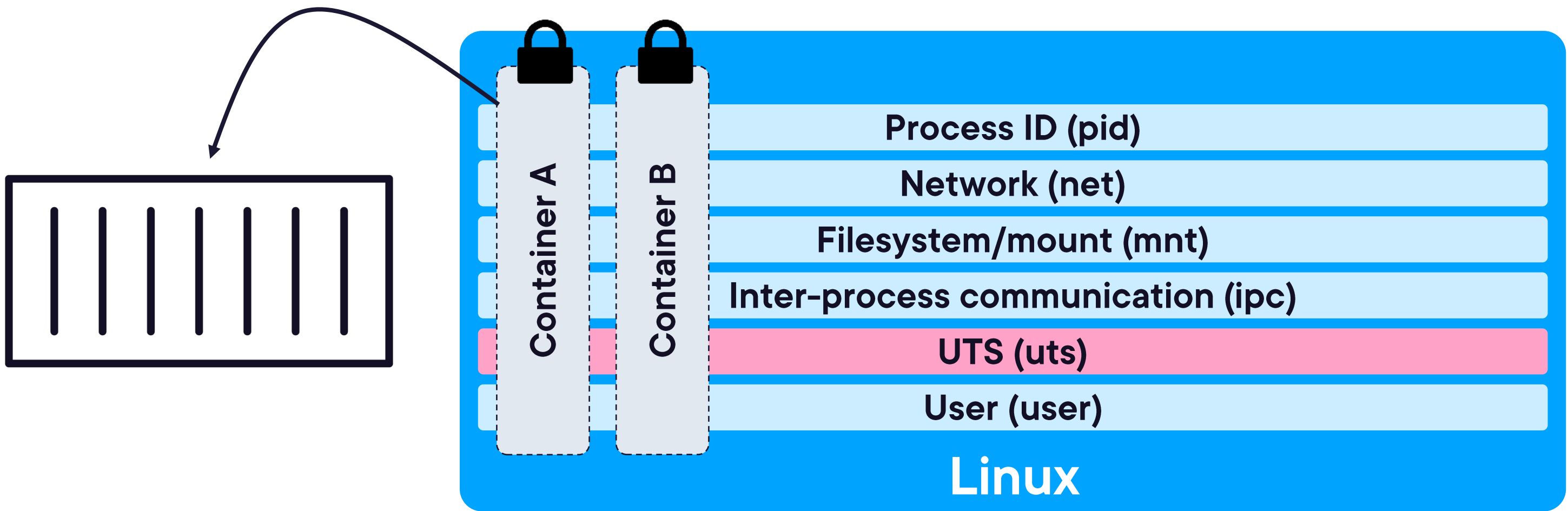




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

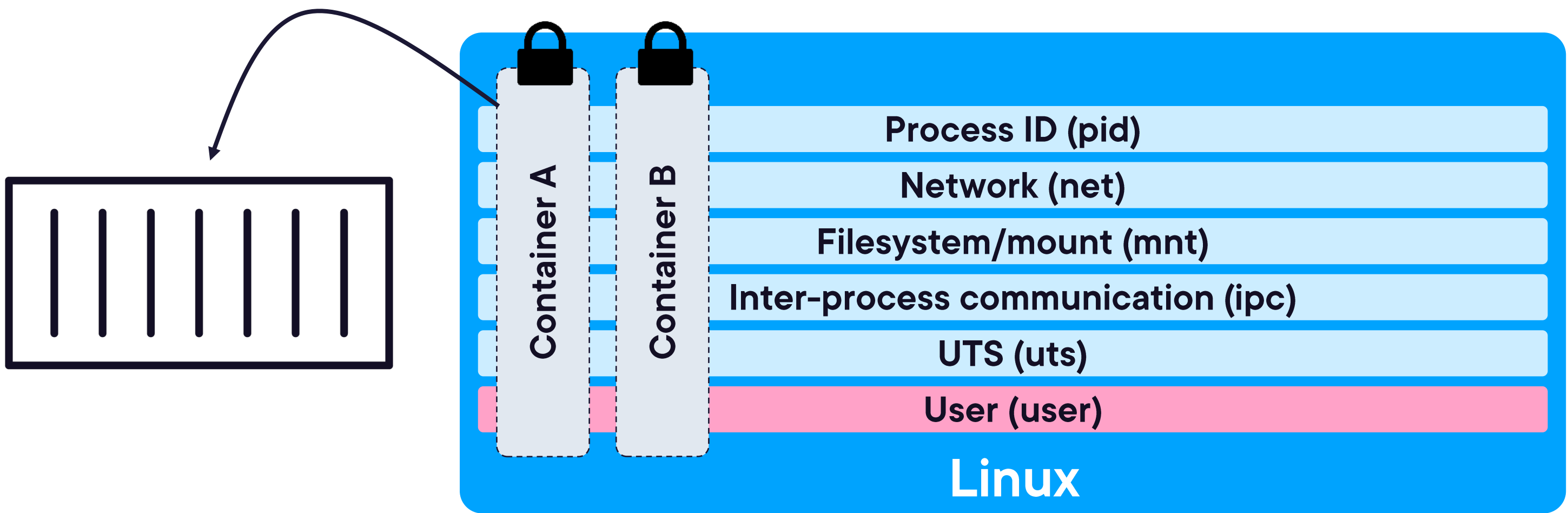




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

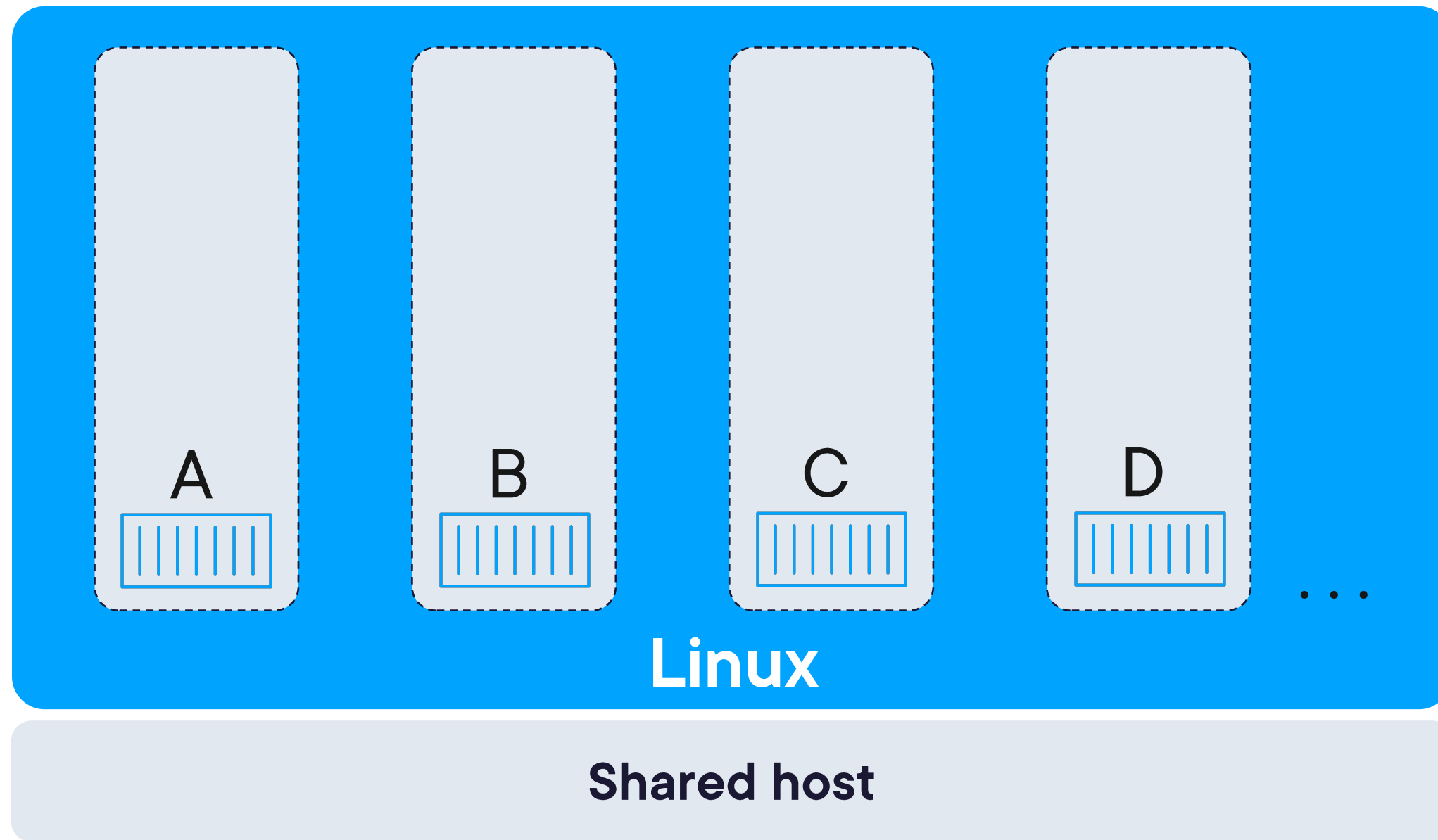




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

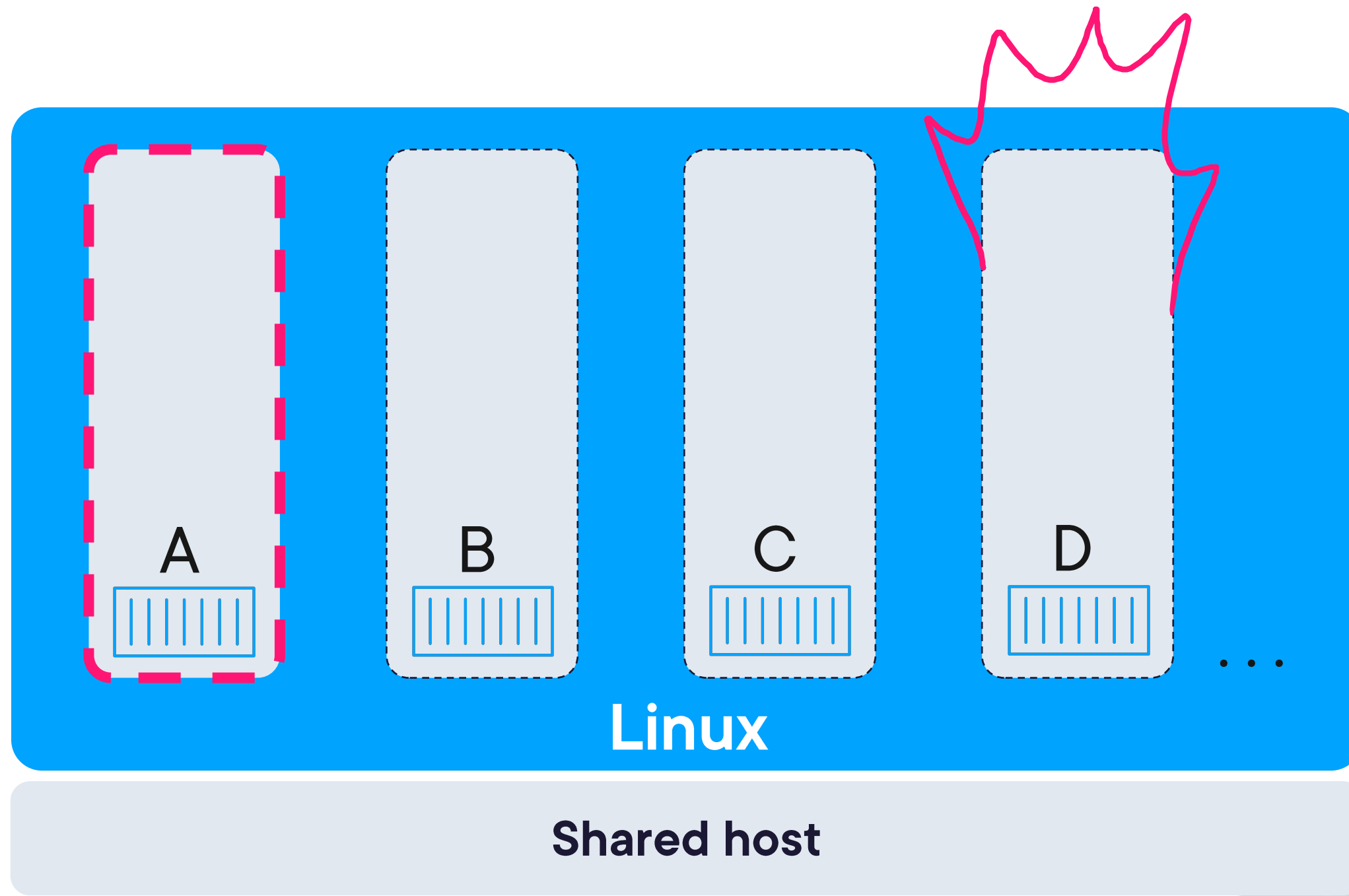




**Namespaces**  
All about isolation

**Control Groups**  
All about limits

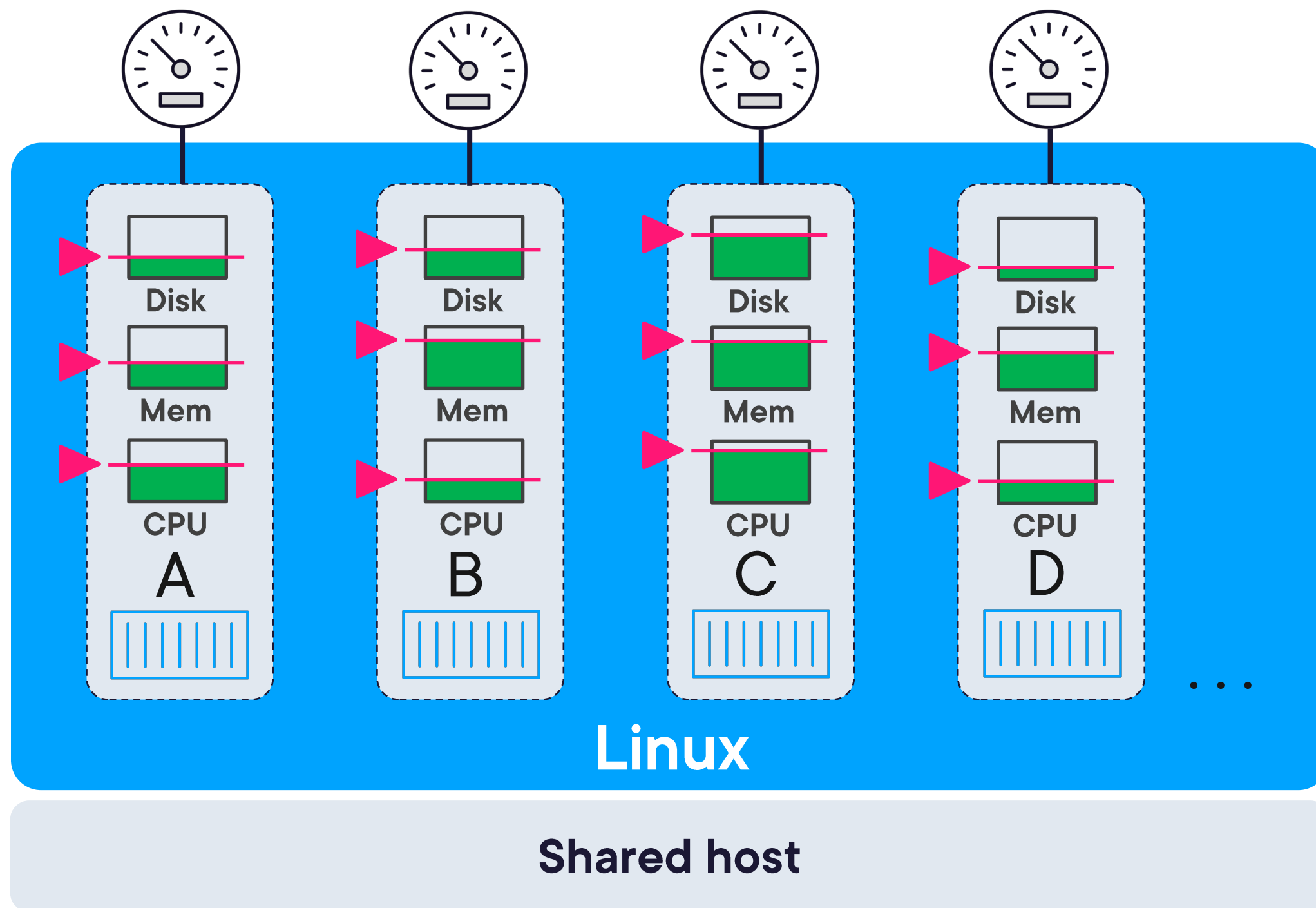




**Namespaces**  
All about isolation

**Control Groups**  
All about limits





**Namespaces**

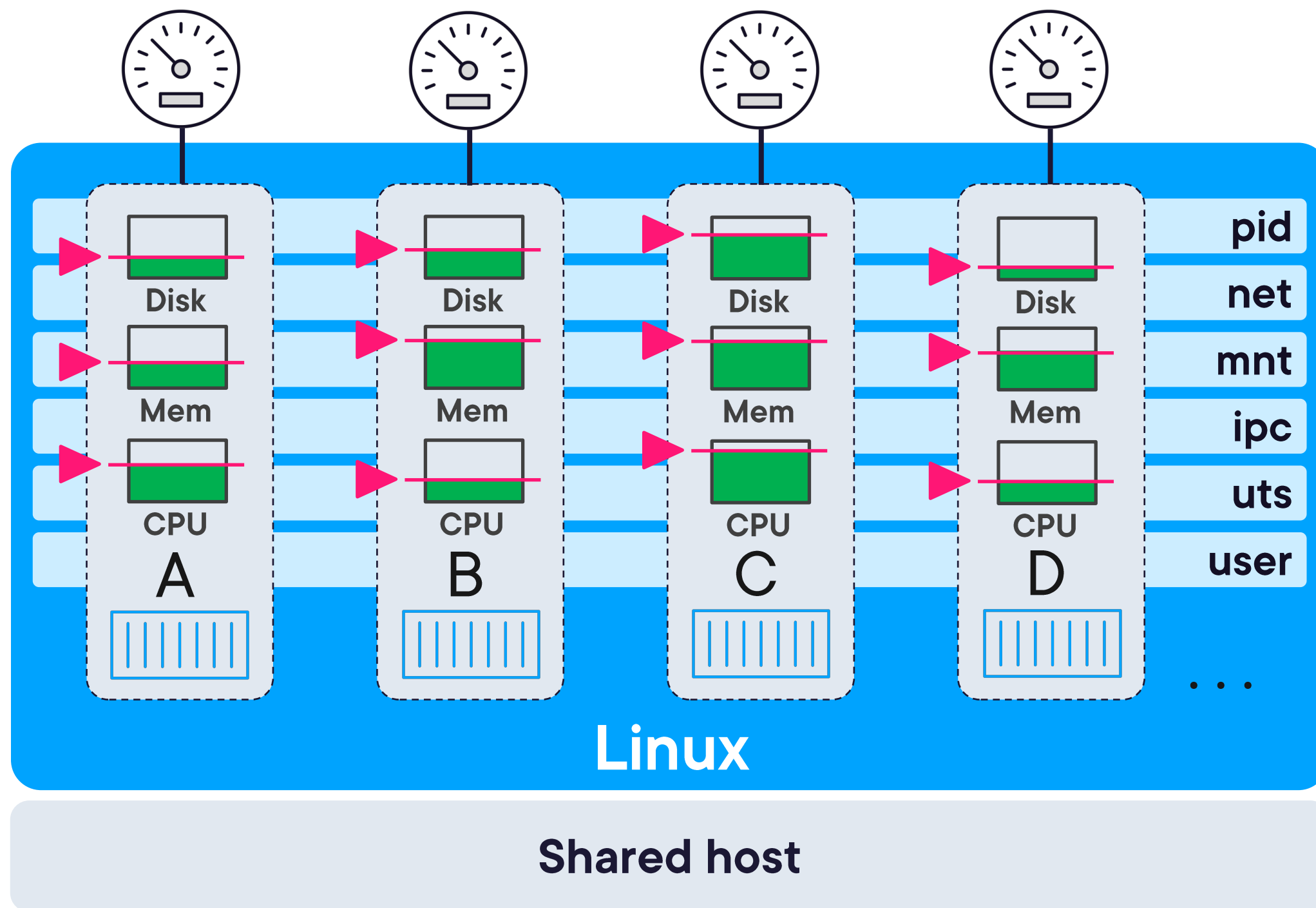
All about isolation

**Control Groups**

All about limits



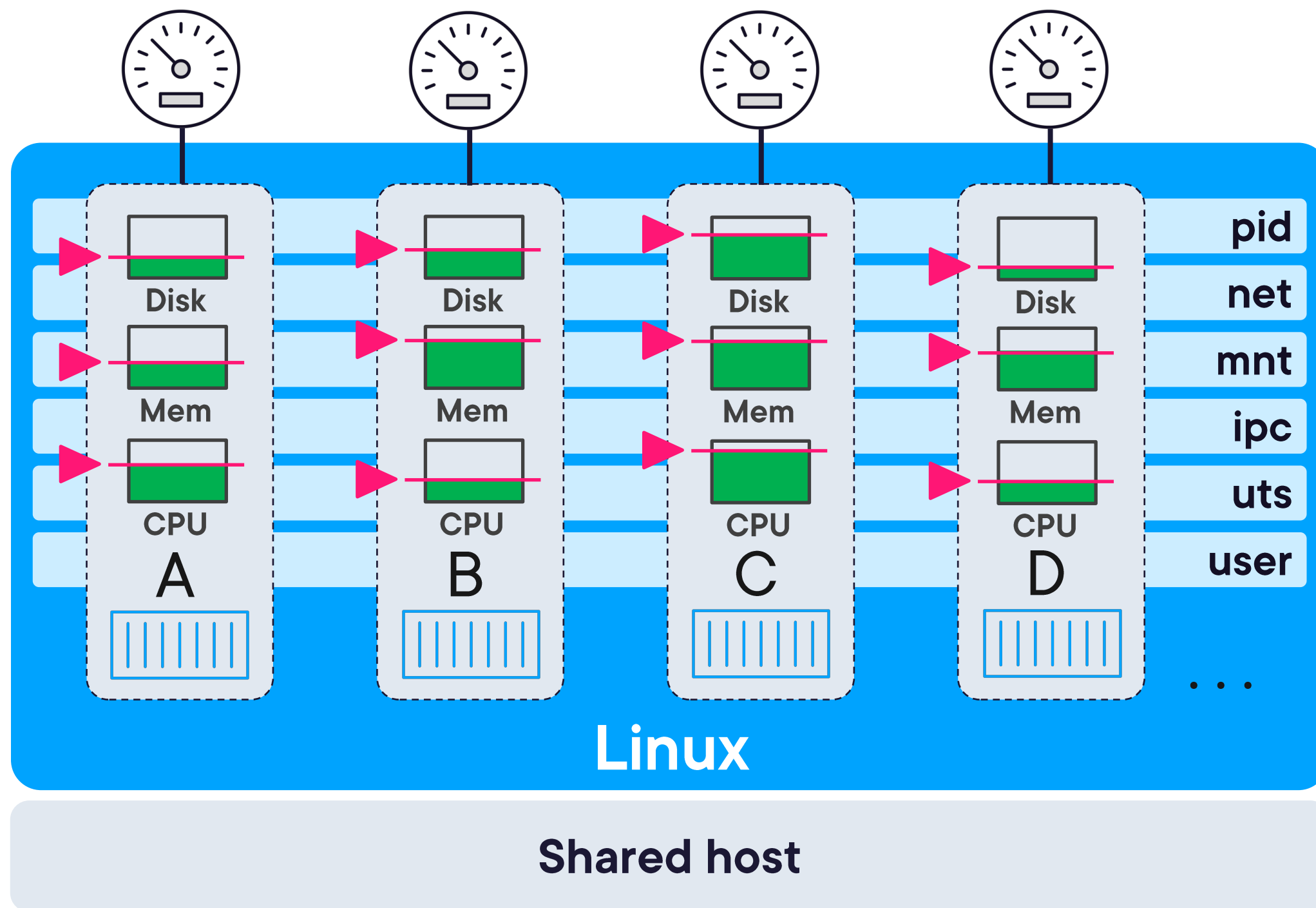




**Namespaces**  
All about isolation

**Control Groups**  
All about limits





**Layered mounts**

Copy on Write

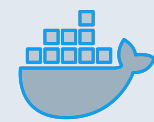
**Namespaces**

All about isolation

**Control Groups**

All about limits





**Docker – The Easy Layer...**



**Layered mounts**  
Copy on Write

**Namespaces**  
All about isolation

**Control Groups**  
All about limits

...

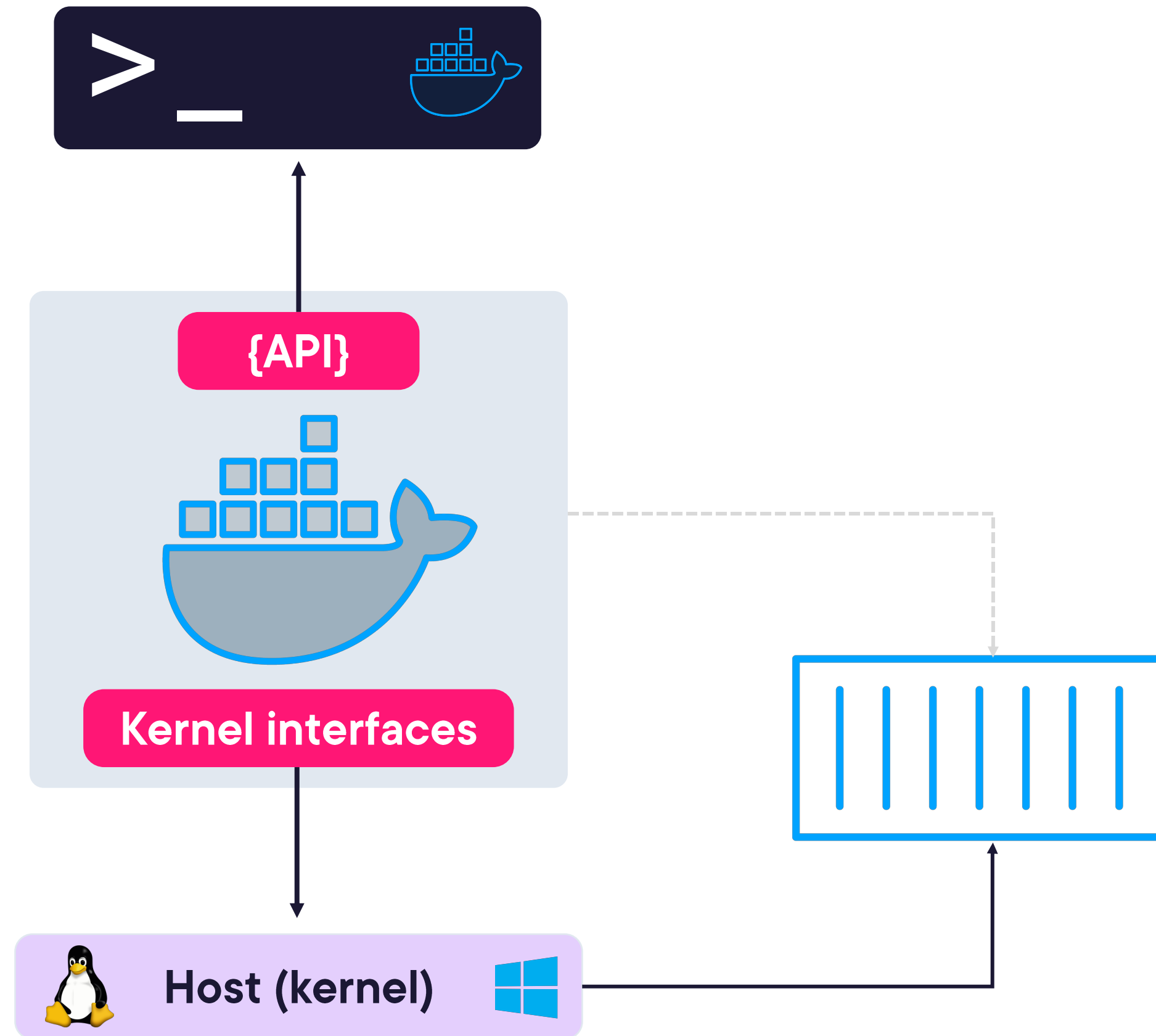


**Up Next:**

# **Docker Magic**

---







dotCloud



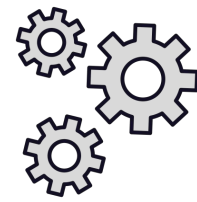


dotCloud



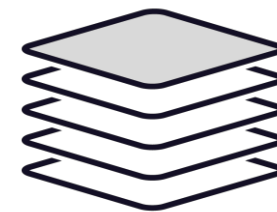


# dotCloud



LXC

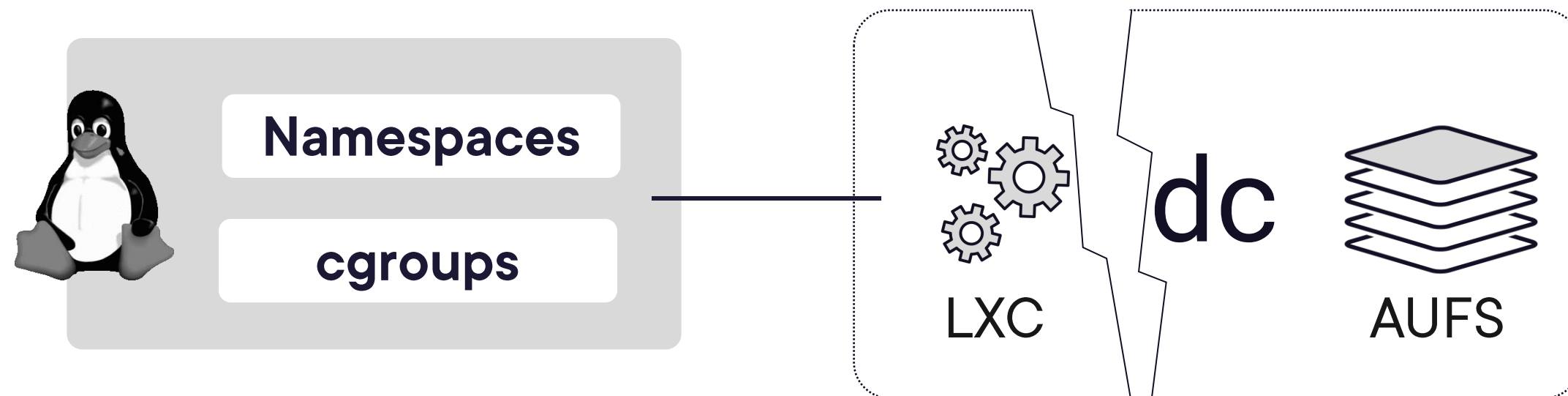
dc

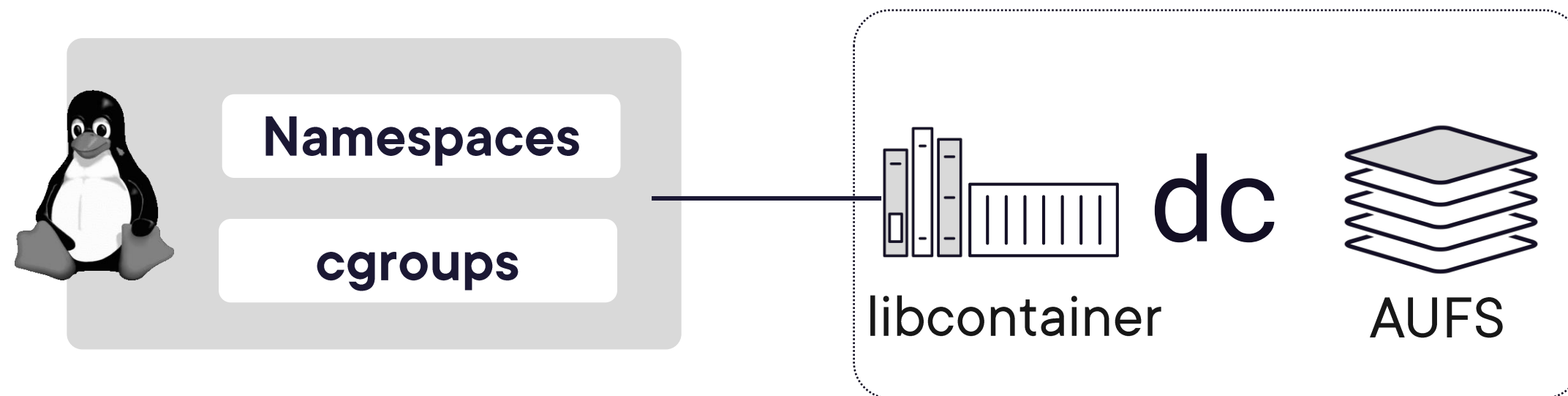


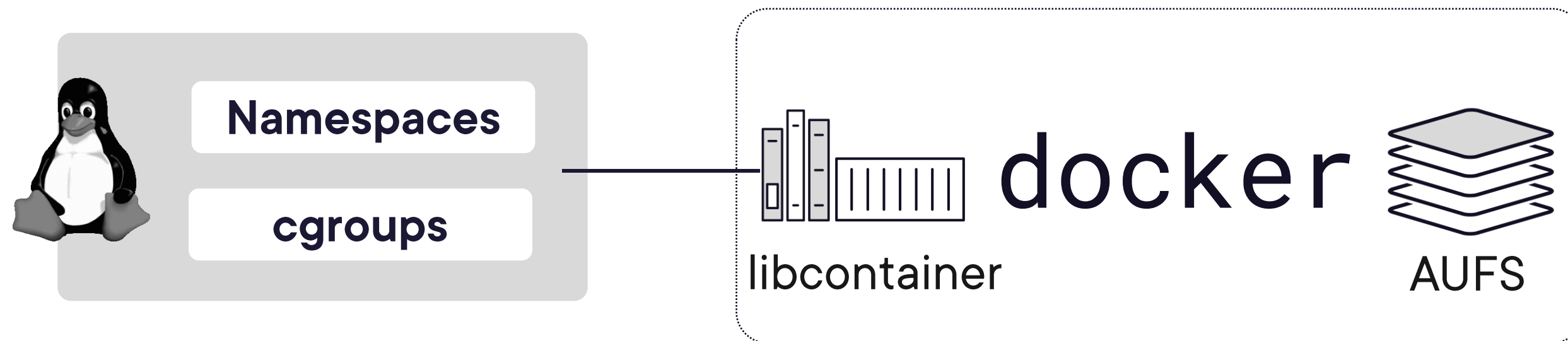
AUFS

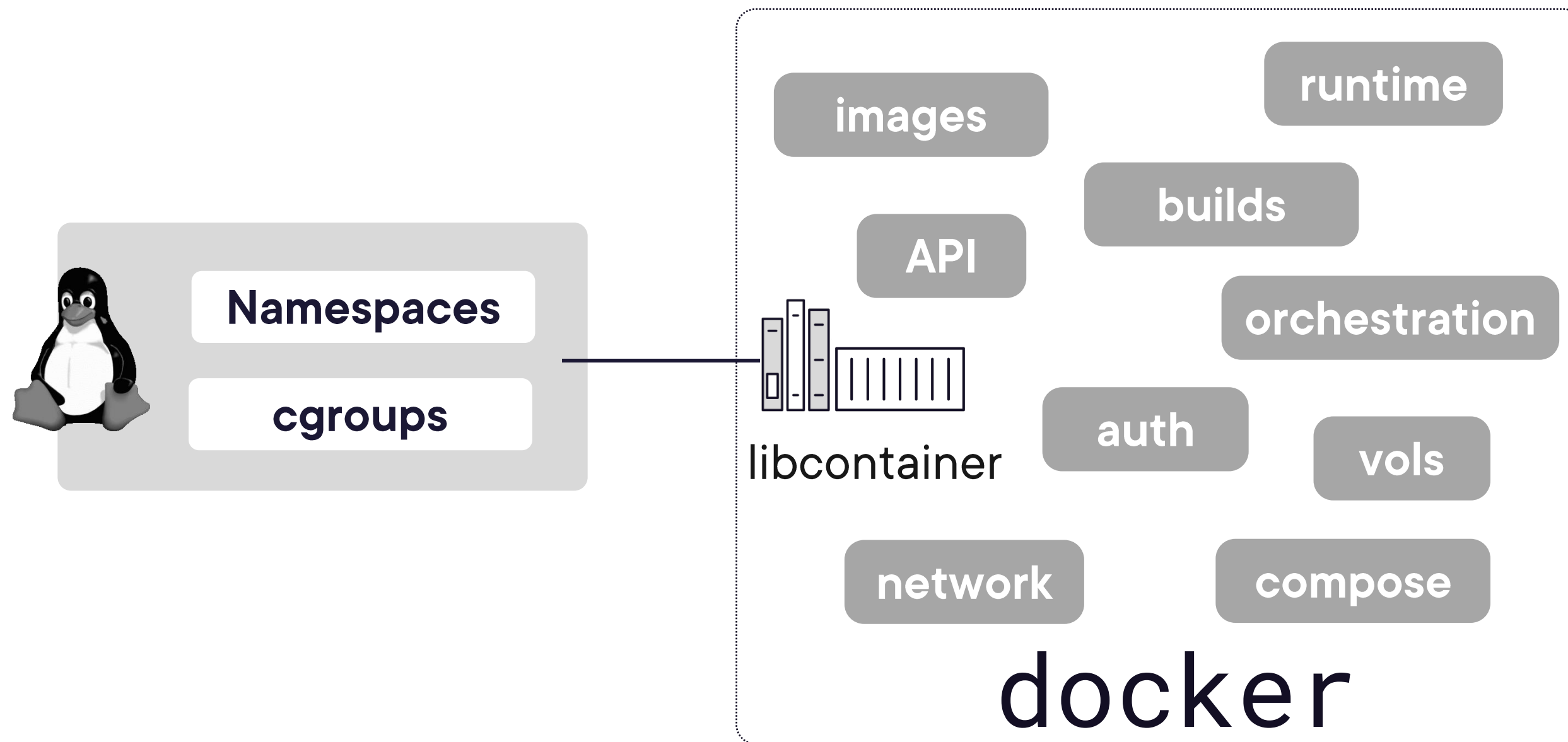


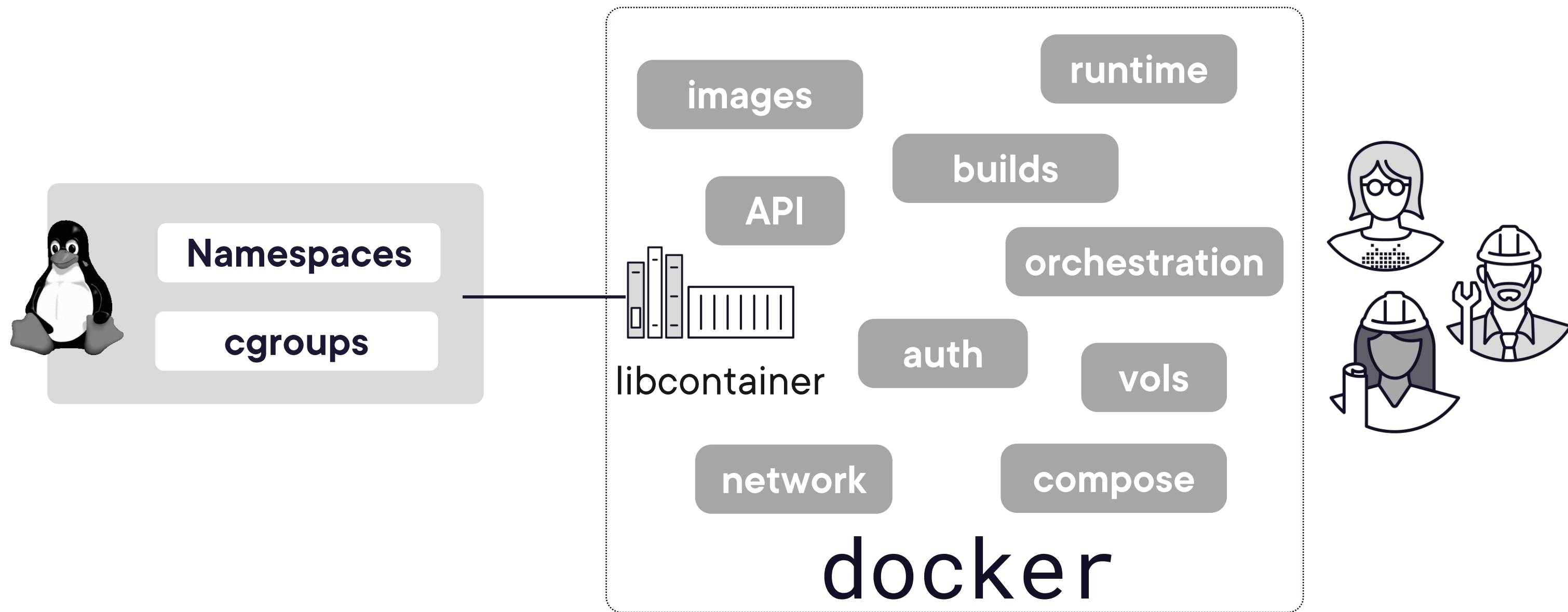


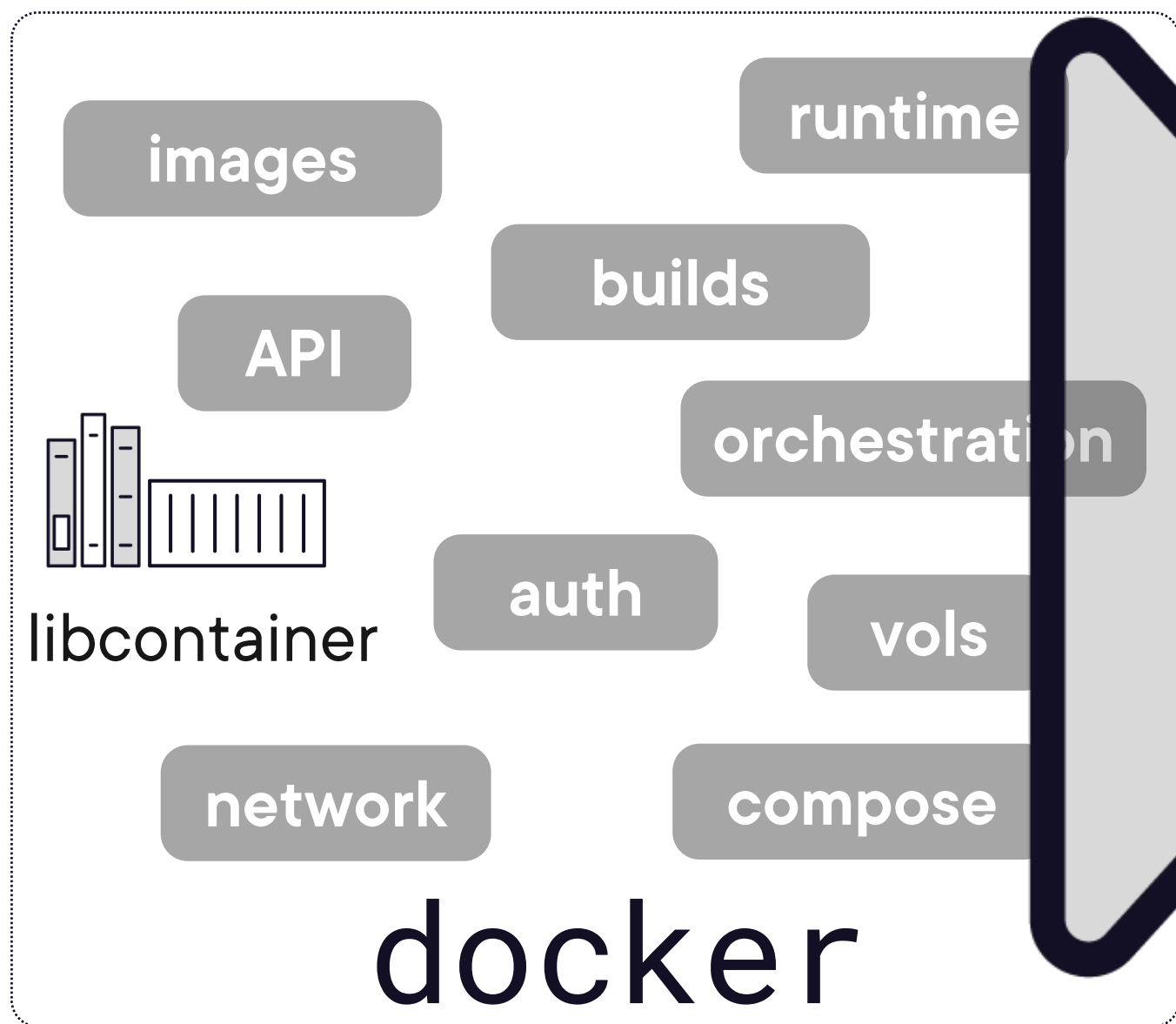












OPEN CONTAINER  
INITIATIVE

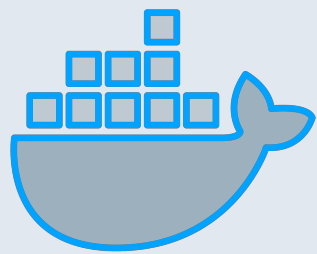
Image spec

Runtime spec



\$ docker run 

{REST} POST /vX.X/containers/create HTTP/1.1



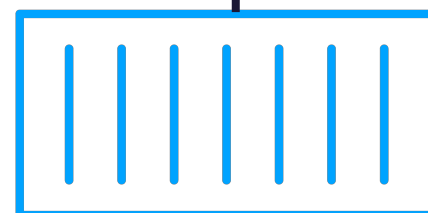
daemon

gRPC {API} client.NewContainer(context, ...

container 

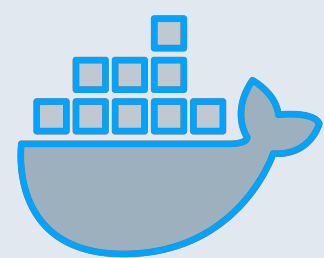
Shim

 RUNC



\$ docker run 

{REST} POST /vX.X/containers/create HTTP/1.1



daemon

gRPC {API} client.NewContainer(context, ...

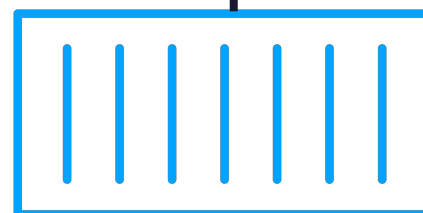
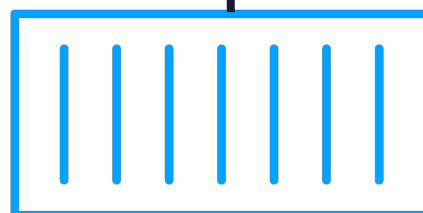
container 

Shim

Shim ...

 runc

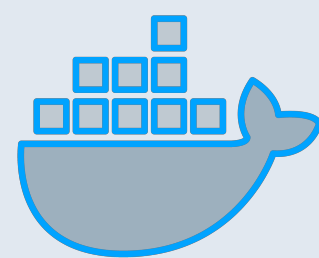
 runc





```
$ docker run
```

{REST} POST /vX.X/containers/create HTTP/1.1



daemon

gRPC {API} client.NewContainer(context, ...



**CLOUD NATIVE**  
COMPUTING FOUNDATION

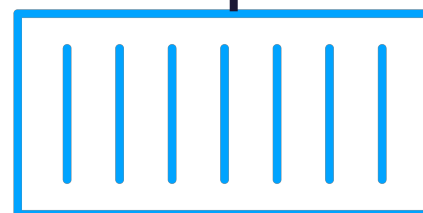
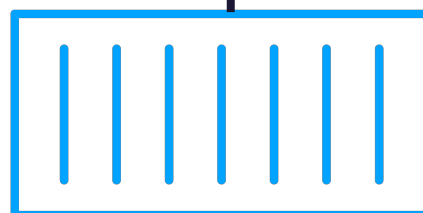
containerd

Shim

Shim ...

 runc

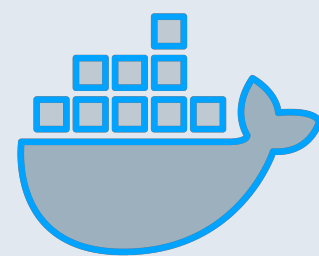
 runc



**OPEN** CONTAINER  
INITIATIVE



\$ docker run 



daemon

Docker API,  
Image management...



**CLOUD NATIVE**  
COMPUTING FOUNDATION

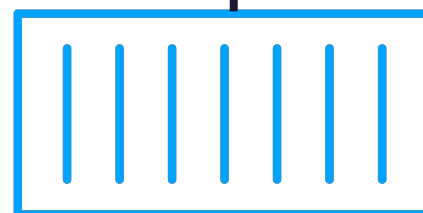
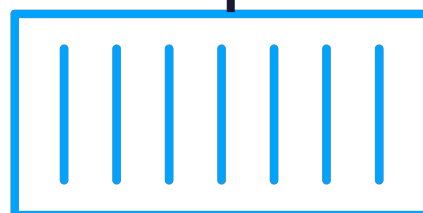
container 

Shim

Shim ...

 RUNC

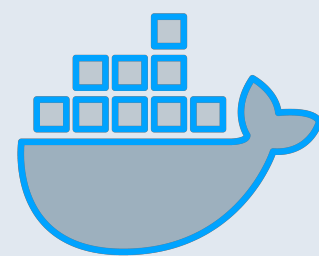
 RUNC



**OPEN** CONTAINER  
INITIATIVE



\$ docker run 



daemon

Docker API,  
Image management...



CLOUD NATIVE  
COMPUTING FOUNDATION

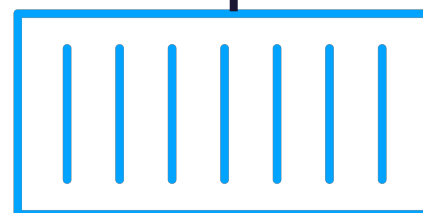
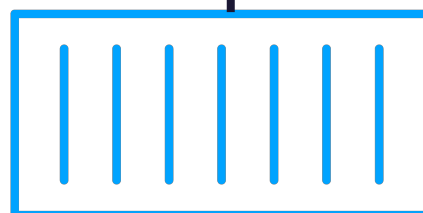
container 

Shim

Shim ...

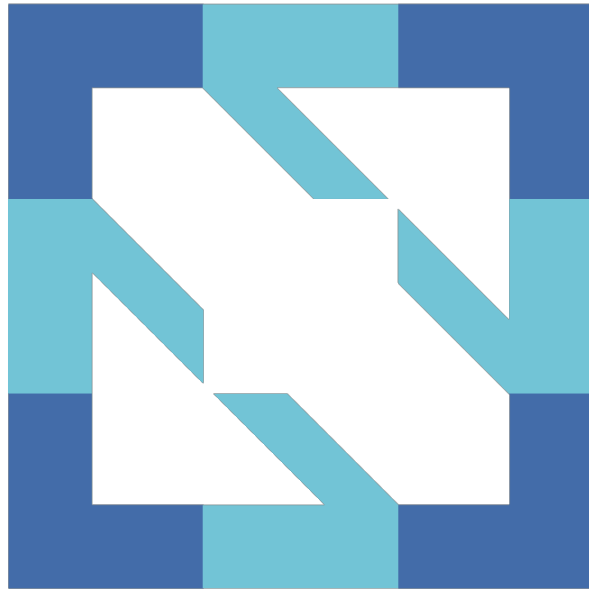
 RUNC

 RUNC



OPEN CONTAINER  
INITIATIVE





**CLOUD NATIVE**  
**COMPUTING FOUNDATION**



**OPEN** **CONTAINER**  
**INITIATIVE**

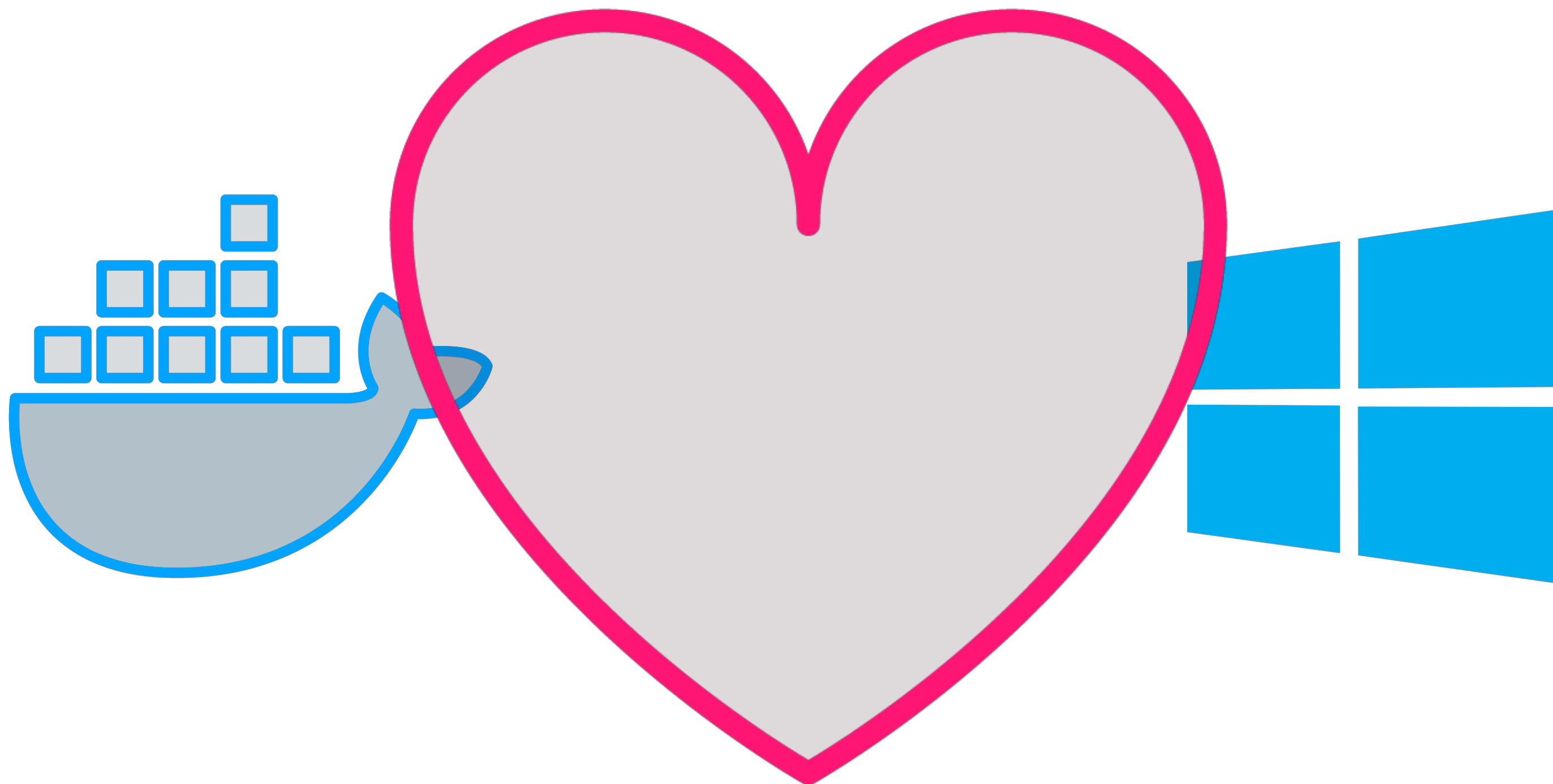








**Up Next:**

# **Containers on Windows**

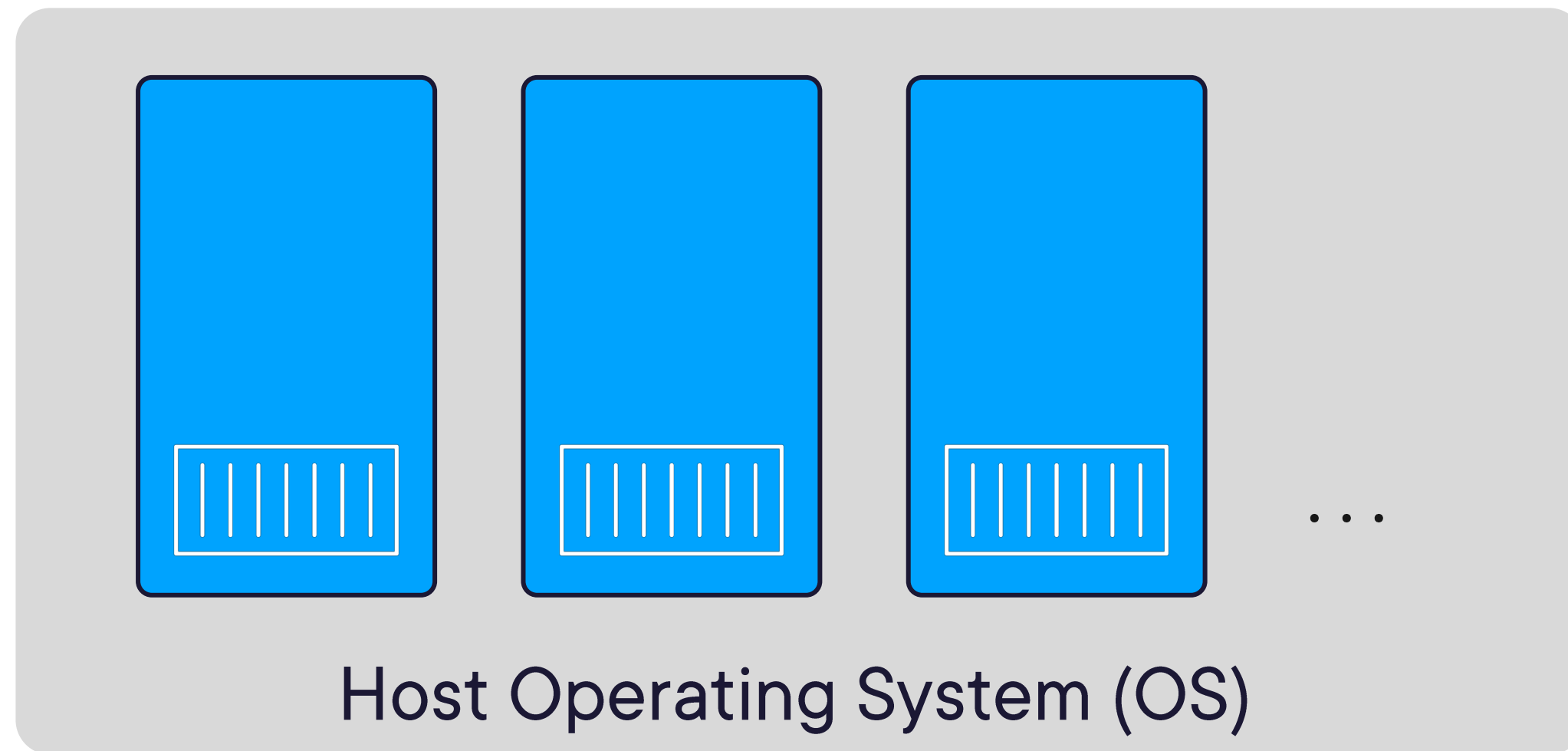
---



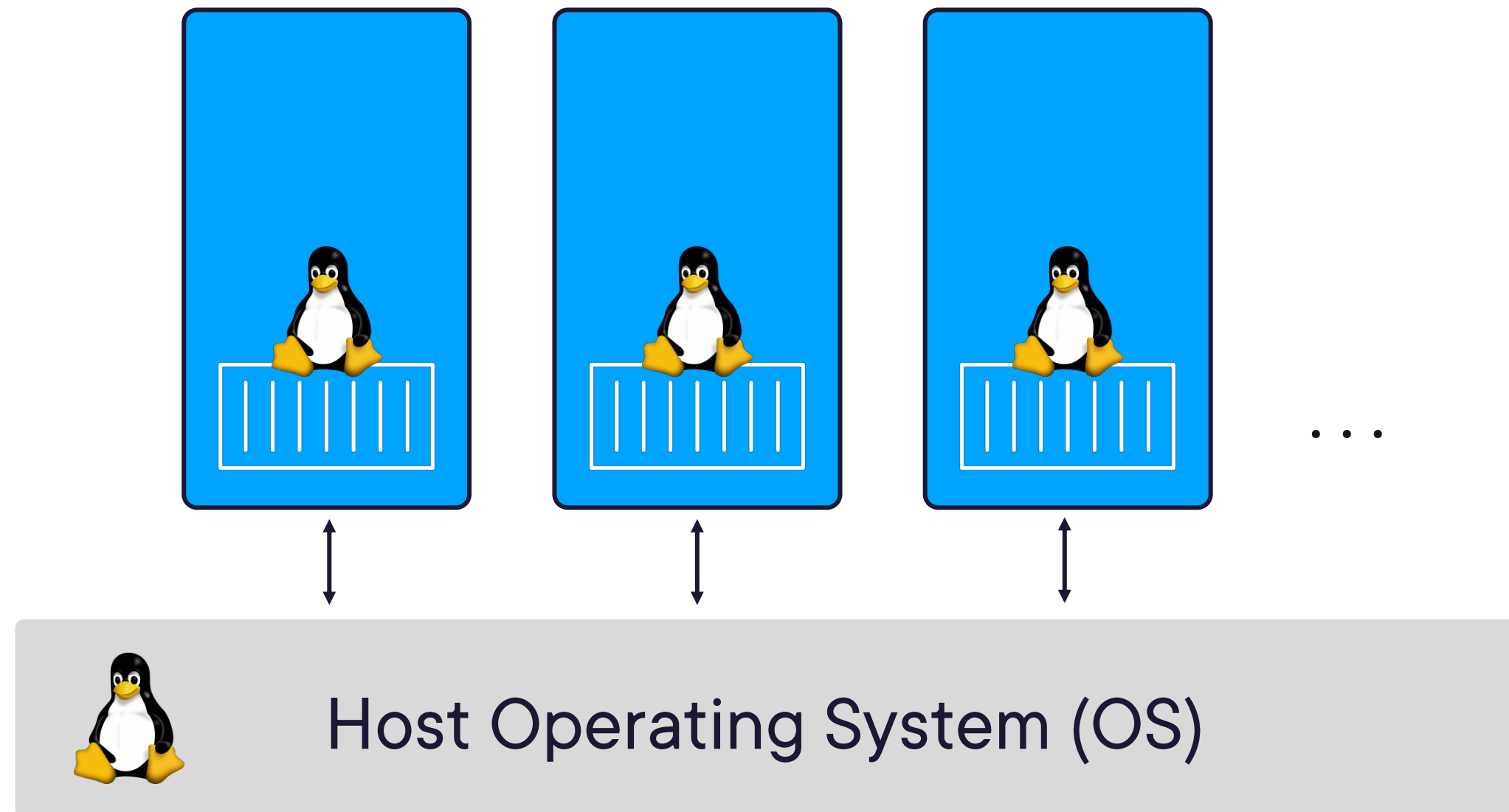


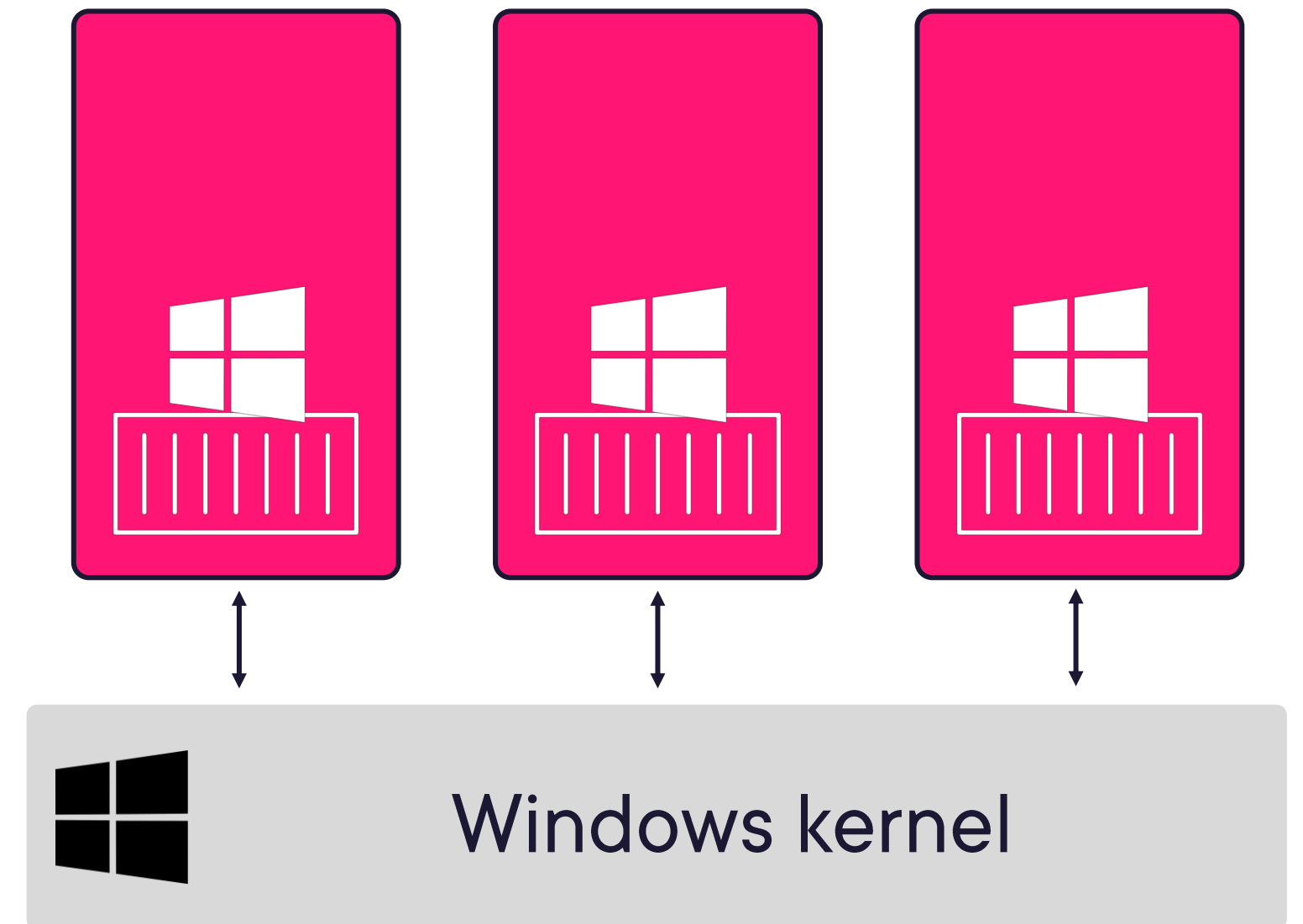
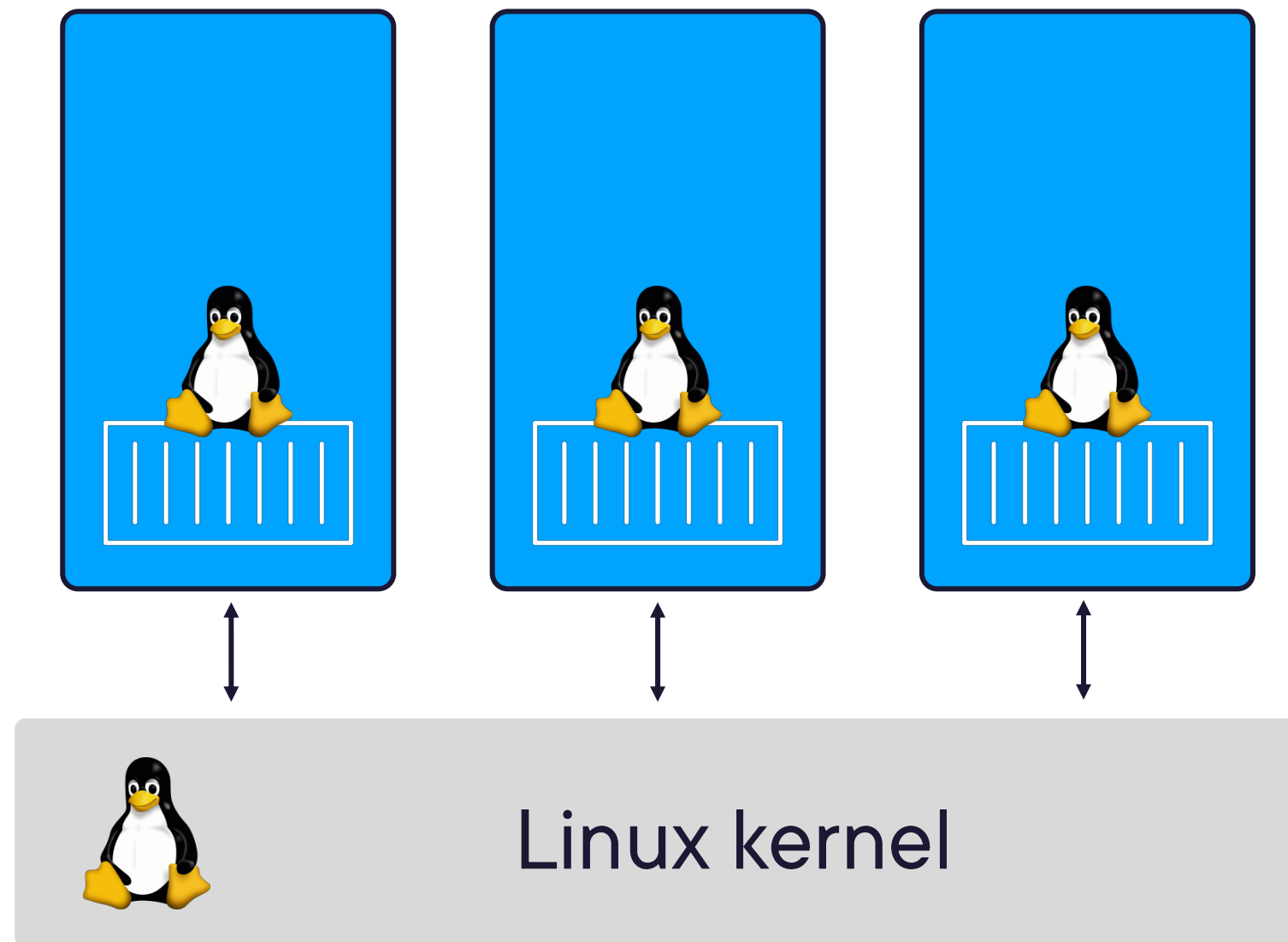
	Docker on macOS	Docker on Linux	Docker on Windows
Linux containers			
Windows containers			

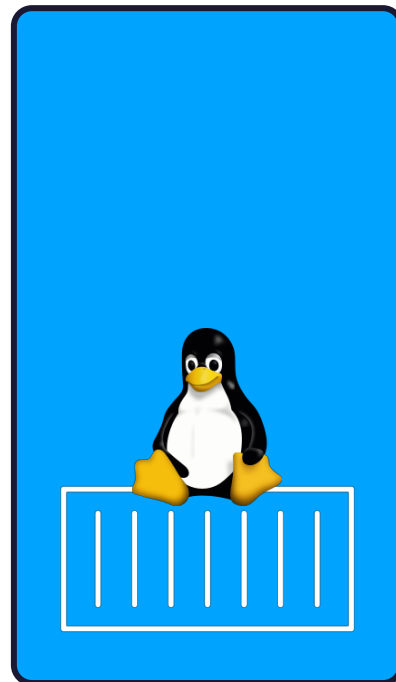
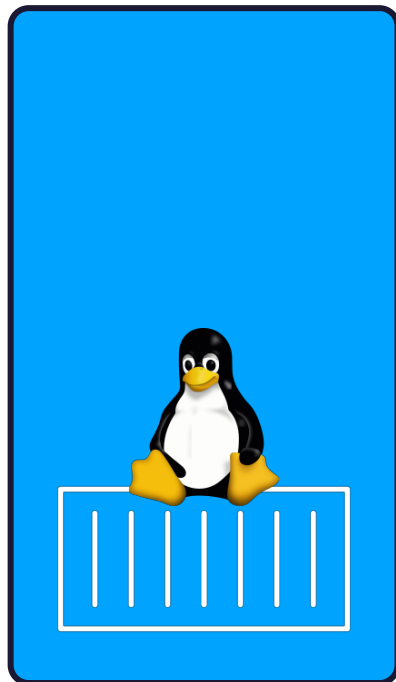
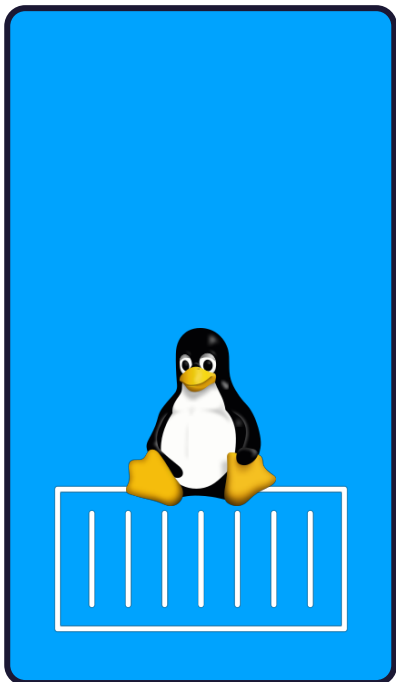
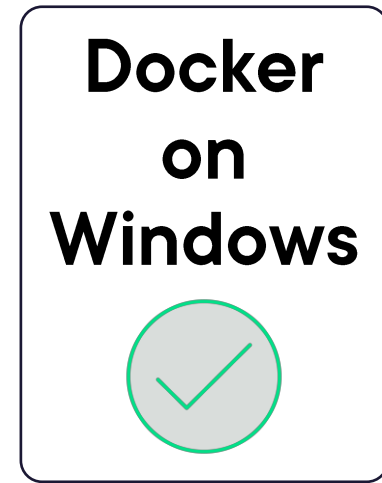
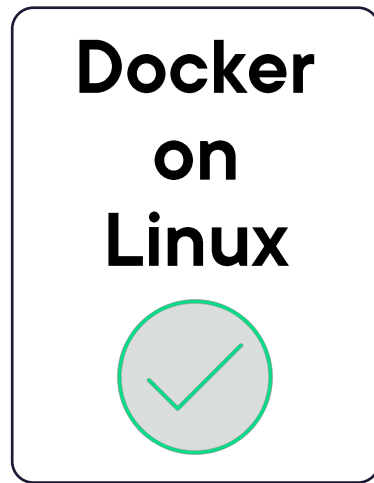
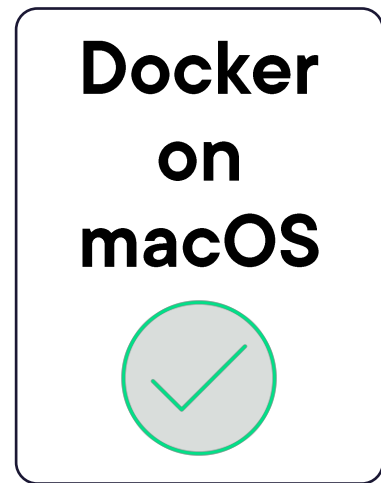




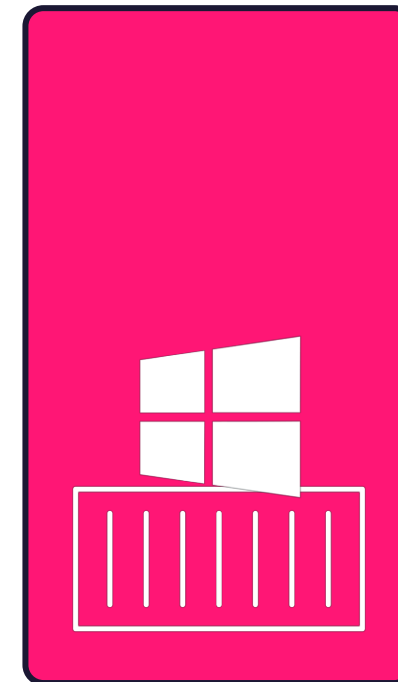
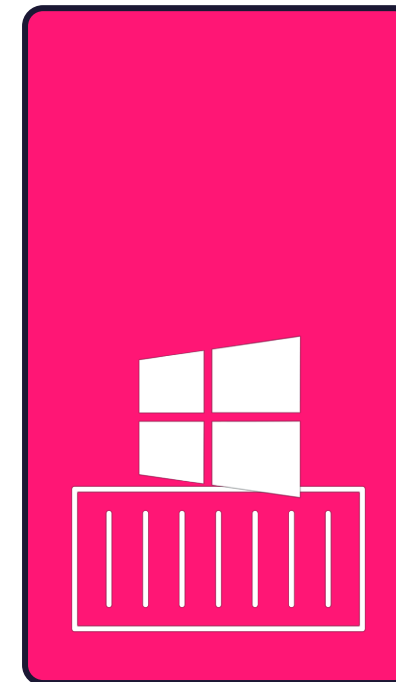
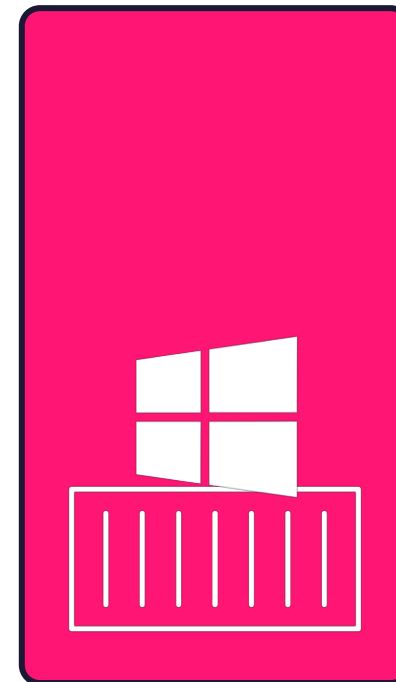
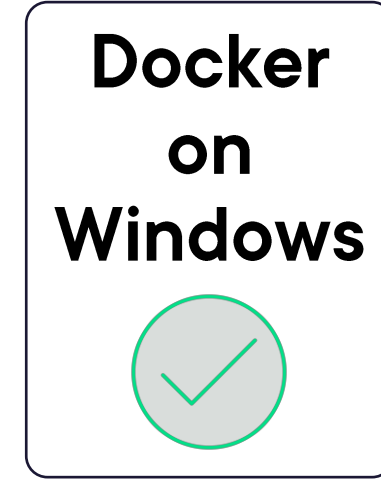








Linux kernel



Windows kernel





## Docker Desktop on Windows

