

# Sessió 1: Contenedors

Xavier Sala Pujolar



Universitat  
de Girona

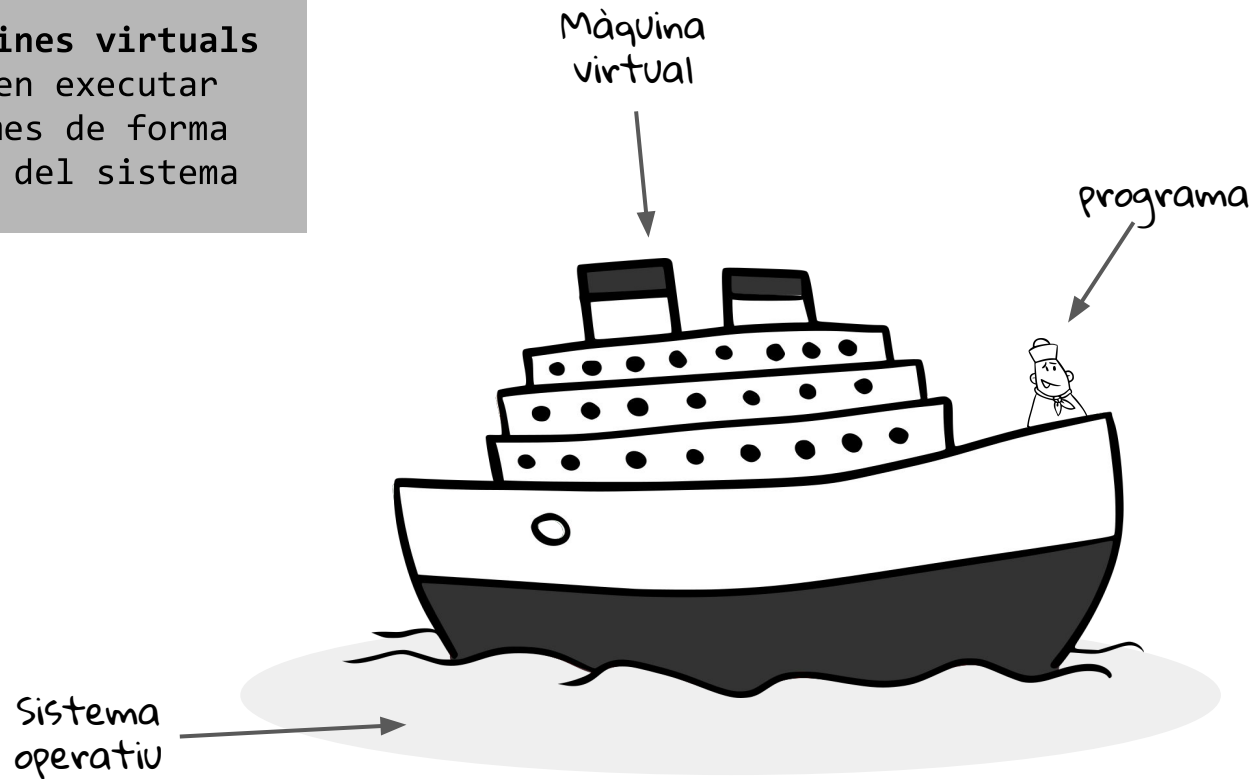
Gener 2021

# Contenidors

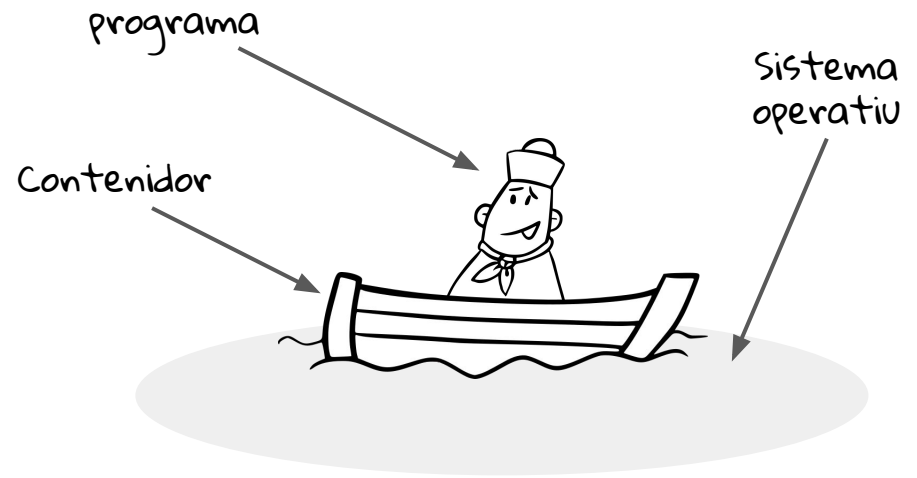
Els contenidors són una tècnica de virtualització dels sistemes operatius que es fan servir per empaquetar aplicacions amb les seves dependències i executar-les en entorns aïllats.

Són una alternativa lleugera a les màquines virtuals.

Les màquines virtuals  
permeten executar  
programes de forma  
aïllada del sistema

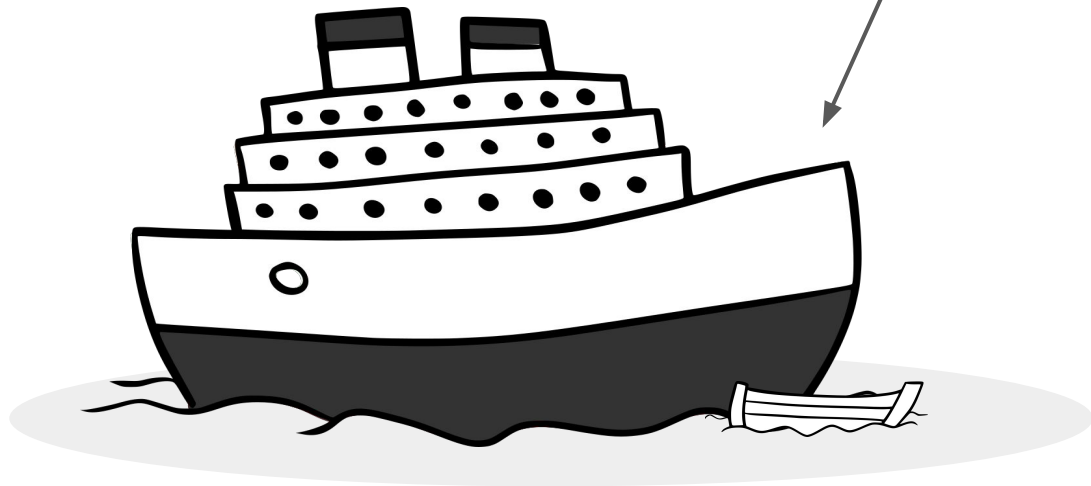


Els **contenidors**  
permeten al sistema  
operatiu **executar**  
**aplicacions** aïllant-les  
de les altres



Els contenidors són  
molt més lleugers que  
les màquines virtuals

Es poden posar  
contenidors dins  
de màquines  
virtuals



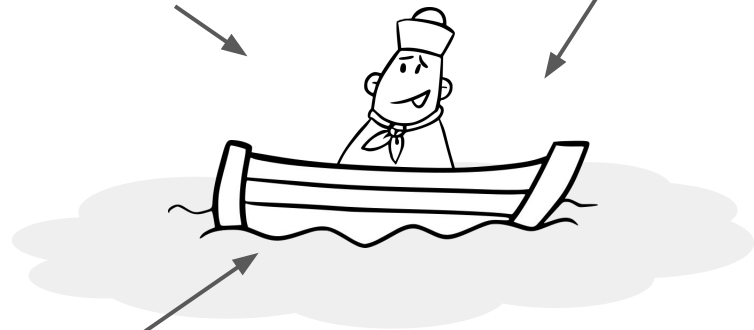
L'objectiu d'un  
contenedor és executar  
un programa de forma  
aïllada

Els executa el nucli del  
sistema operatiu

Cada aplicació té  
la seva pròpia  
versió del sistema

Se'ls hi pot  
limitar l'ús de  
recursos

Tenen el mínim  
necessari per  
funcionar



No poden executar  
contenidors d'altres  
sistemes operatius

Ni entorns gràfics

Ni migració en calent

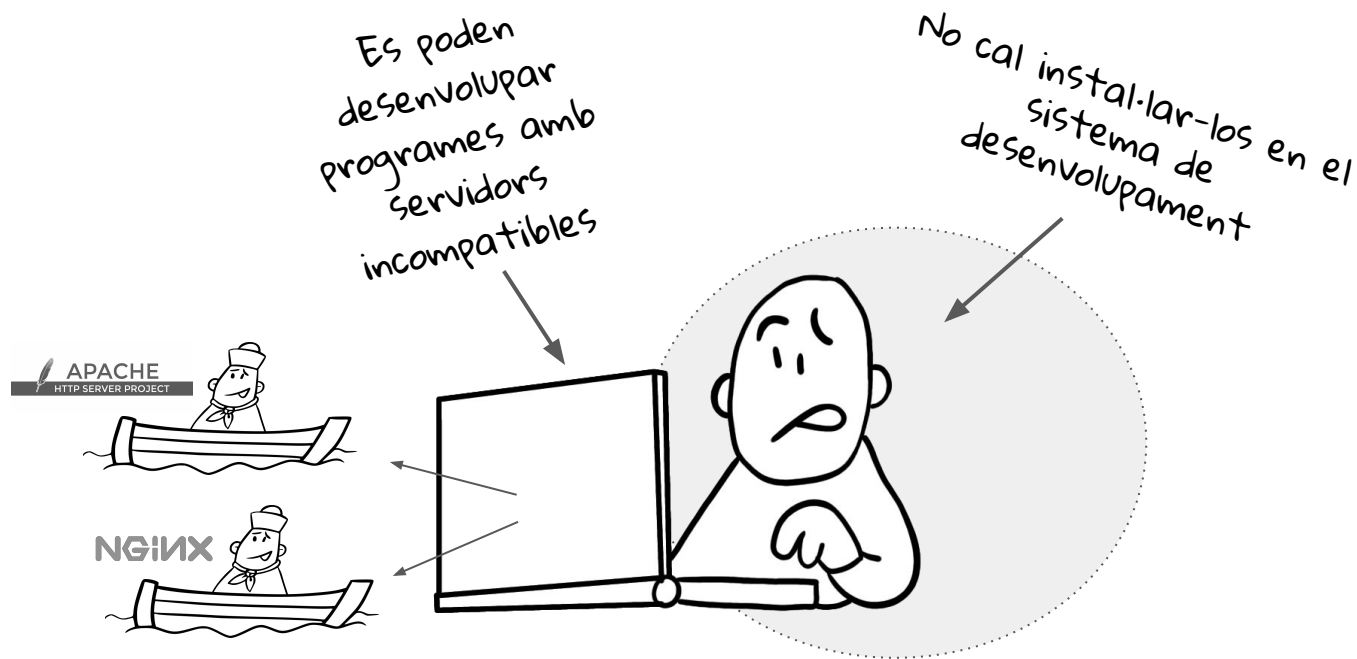
Tampoc contenidors  
d'altres sistemes  
operatius



Linux



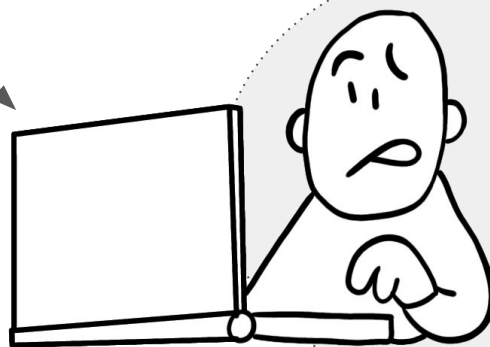
# Què aporten pel desenvolupament?





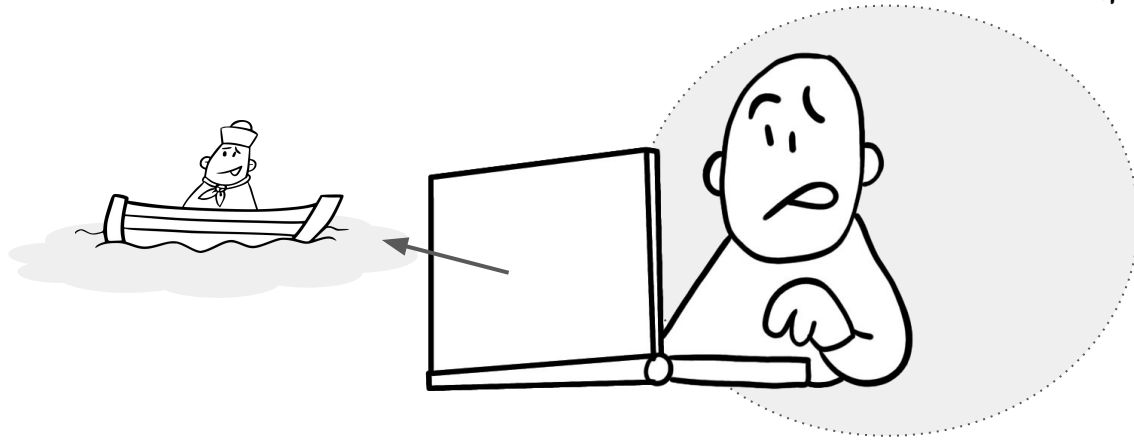
És fàcil duplicar  
l'entorn de  
treball

llibreria,  
compilador, ...



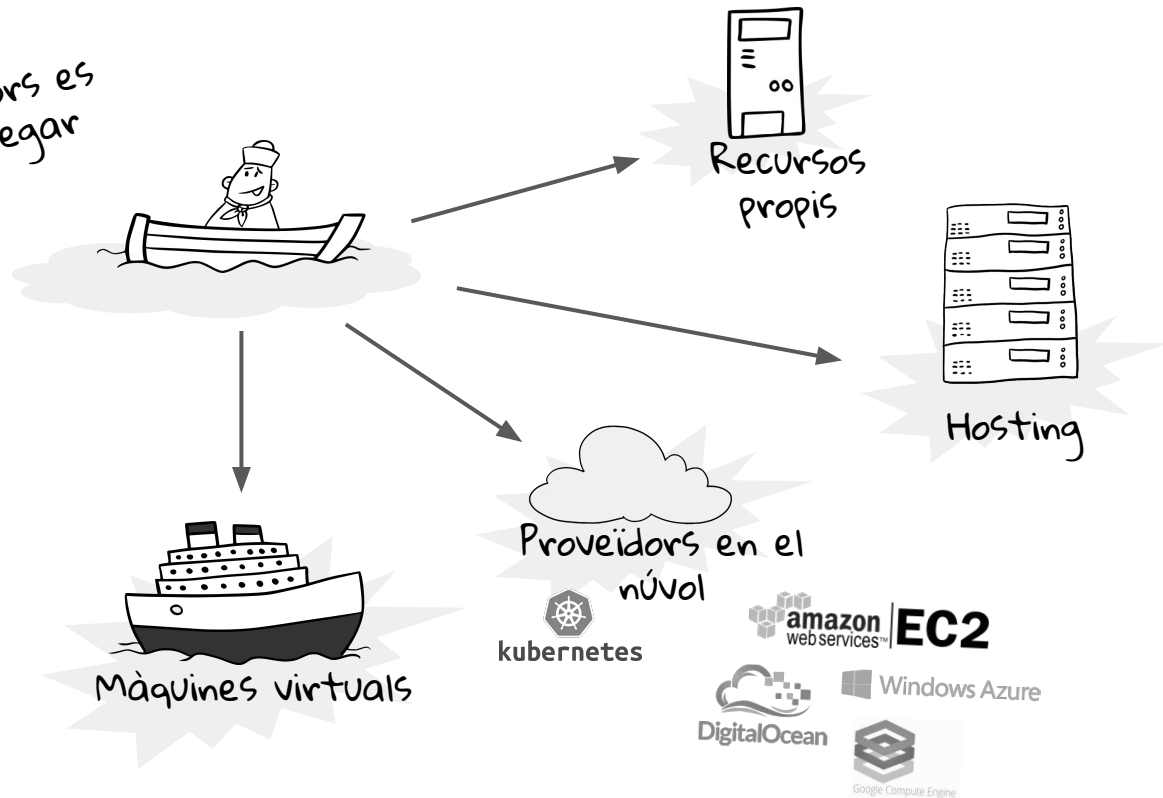
Pots provar en la teva  
màquina l'aplicació en l'entorn  
on funcionarà !

s'ha acabat  
"en el meu ordinador  
funcionava"

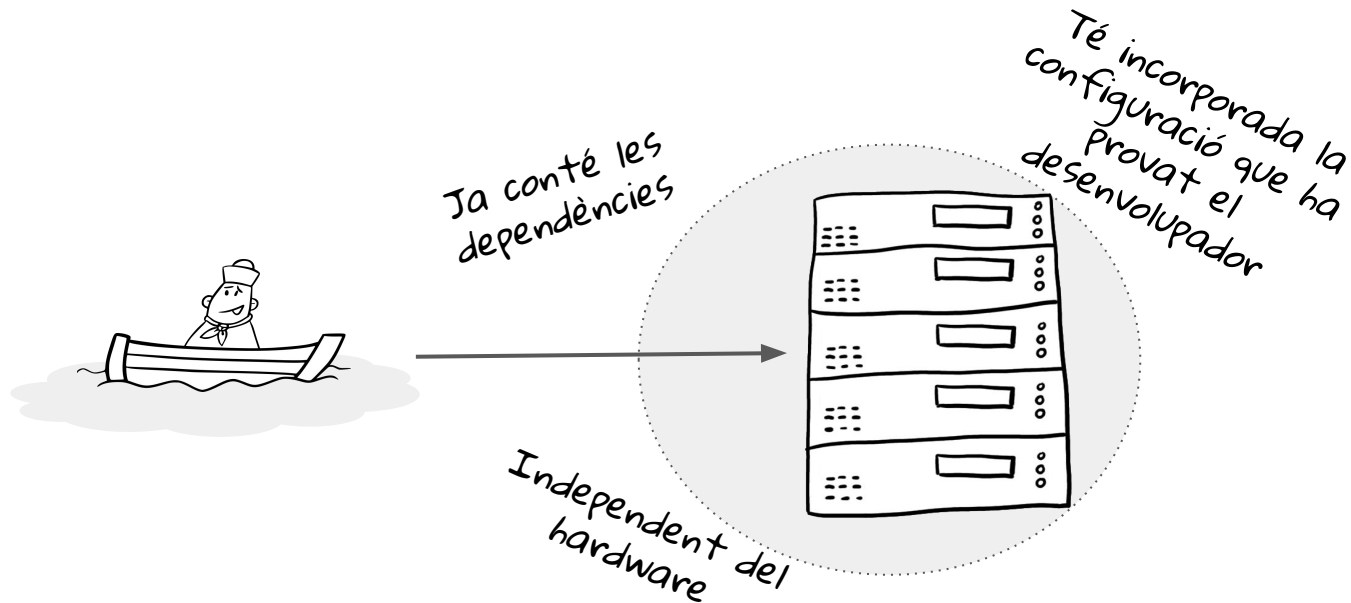


# Què aporten al desplegament?

Els contenidors es  
poden desplegar  
....



Simplifiquen moltíssim el desplegament



# DevOps

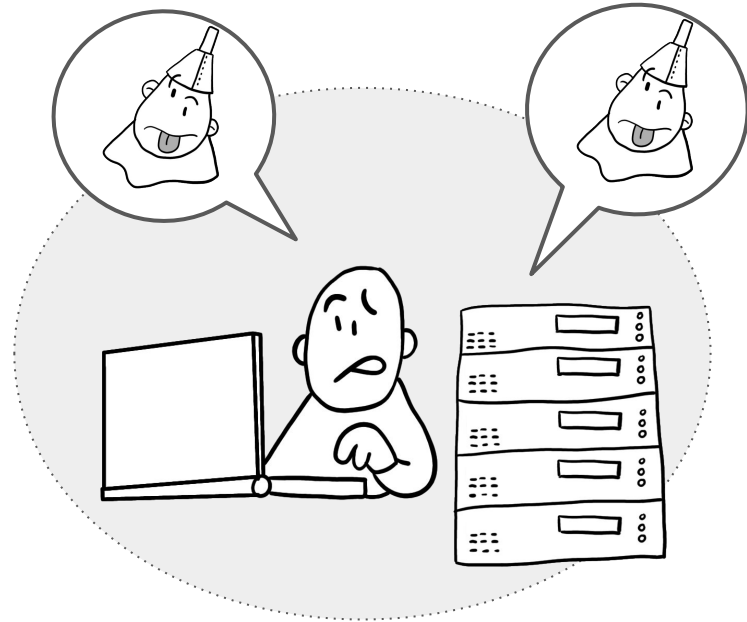
---

Integrar desenvolupament  
amb sistemes

Definir la infraestructura  
com a codi

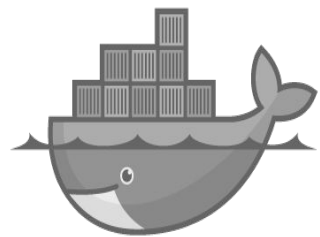
Infraestructura  
repetible

Automatitzable  
(sense errors humans)



# Tecnologies de contenidors

---



docker

Es la referència

No és la única  
tecnologia de  
contenidors



 rkt

 LXC



podman



Hi ha un estàndard  
d'interoperabilitat

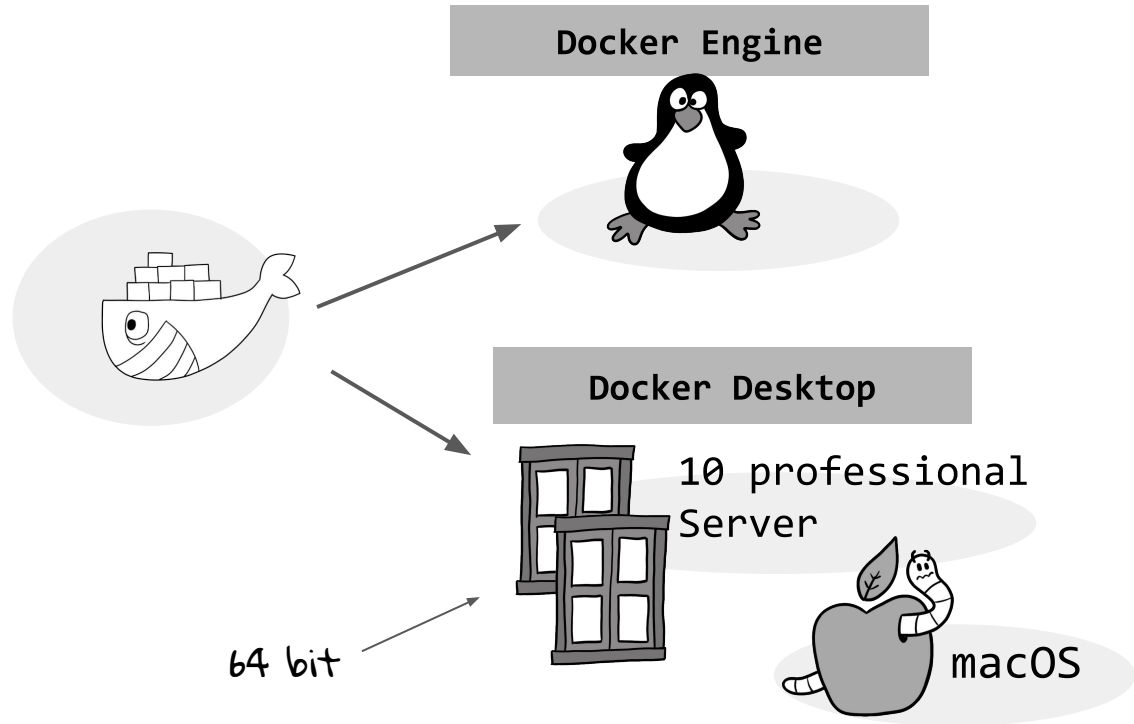
Open Container Initiative



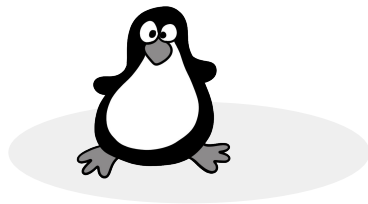
P1: Instal·lació

<https://github.com/utrescu/curs-docker-UdG/tree/day1/day1-1>

# Instal·lació

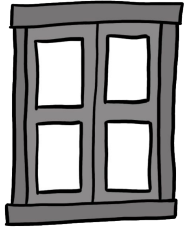






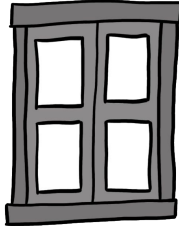
## Linux

- Només instal·la el motor
- S'ha d'afegir l'usuari al grup del servidor



## Professional o Server

- Activar les característiques Hyper-V i Containers Windows
- Suport per contenidors Windows i Linux



## Home



- Activar WSL (Windows subsystem for Linux)
- Suport per contenidors Linux

<https://docs.microsoft.com/en-us/windows/wsl/install-win10>



P2: Vu11 un Ubuntu!

<https://github.com/utrescu/curs-docker-UdG/tree/day1/day1-2>

```
$ docker run ubuntu ls
```

Localitza la  
imatge "ubuntu"  
amb etiqueta  
"latest"

Crea el  
contenedor  
amb nom  
aleatori



Executa ls dins del  
contenedor

```
bin  
boot  
dev  
etc  
...  
...  
usr  
var
```

```
$ docker run -ti ubuntu bash
```

Localitza la  
imatge "ubuntu"  
amb etiqueta  
"latest"

Crea el  
contenedor  
amb nom  
aleatori



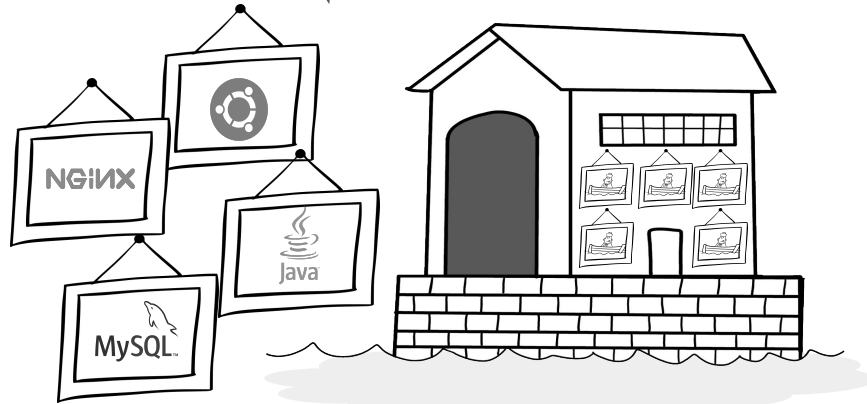
festival\_mari

Interactiu (-ti)

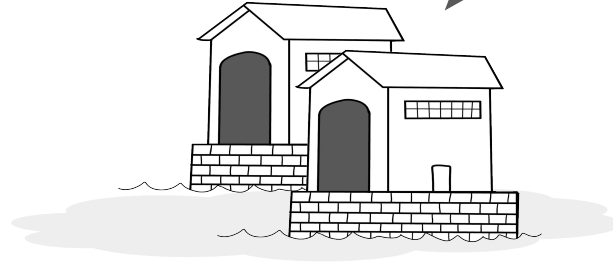
1f34 \$

Executa **bash** dins  
del contenidor

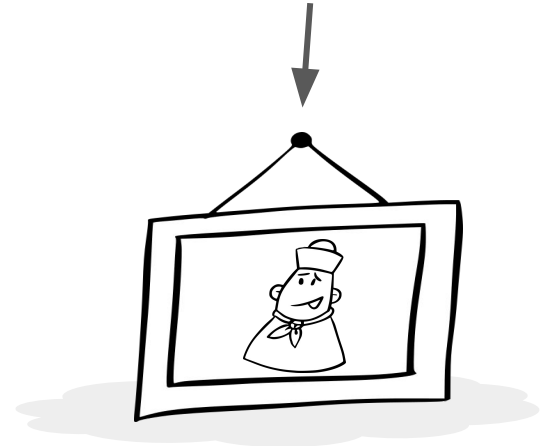
Per defecte es  
descarreguen del  
**Docker Hub**



Es poden fer  
repositoris públics i  
privats



Es poden crear





ubuntu ☆

Docker Official Images

Ubuntu is a Debian-based Linux operating system based on free software.

↓ 18+

Container Linux 386 ARM 64 x86-64 IBM Z PowerPC 64 LE ARM Base Images Operating Systems

Official image

Linux - x86 ( latest )

Copy and paste to pull this image

`docker pull ubuntu`



[View Available Tags](#)

Description

Reviews

Tags

## Quick reference

- **Maintained by:** Canonical and Tianon (Debian Developer)
- **Where to get help:** the Docker Community Forums, the Docker Community Slack, or Stack Overflow

## Supported tags and respective Dockerfile links

- 18.04, bionic-20201119, bionic
- 20.04, focal-20201106, focal, latest
- 20.10, groovy-20201125.2, groovy, rolling
- 21.04, hirsute-20201119, hirsute, devel
- 14.04, trusty-20191217, trusty
- 16.04, xenial-20201030, xenial



P3: Serveis

<https://github.com/utrescu/curs-docker-UdG/tree/day1/day1-3>

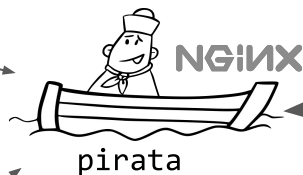


# Executar serveis

```
$ docker run --name pirata -d nginx
```

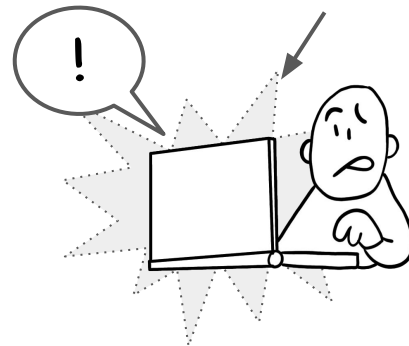
Crea el  
contenedor  
amb nom  
pirata

Executa el servidor  
del contenedor



Localitza la  
imatge "nginx"  
amb versió  
"latest"

Hi podem  
contactar amb  
un navegador





P4: Compilar sense compilador

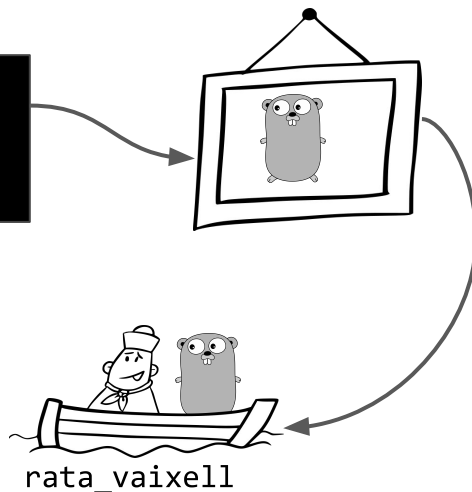
<https://github.com/utrescu/curs-docker-UdG/tree/day1/day1-4>

# Compilar en Go!

```
$ docker run --rm -ti  
-v $PWD:/go  
golang bash
```

Crea el  
contenedor  
amb nom  
aleatori

Comparteix  
el directori



Localitza la  
imatge "Golang"  
amb versió  
"latest"

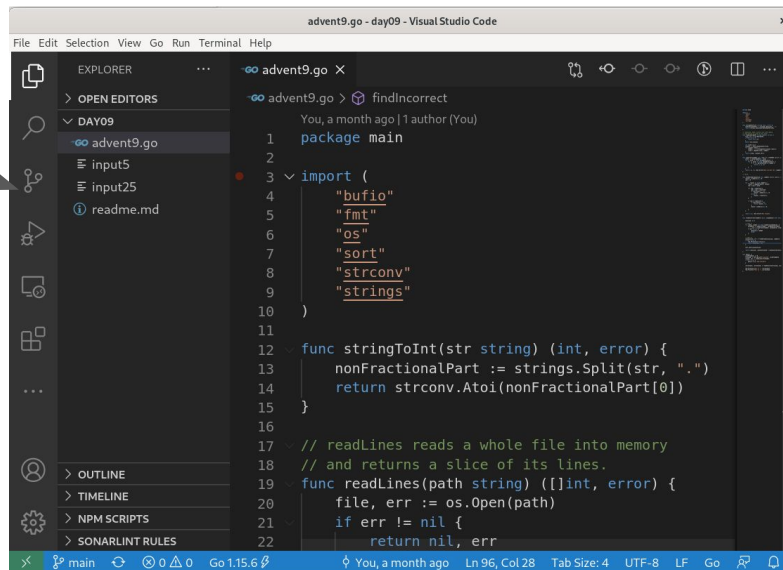
En el bash puc  
compilar i executar

```
4f21 $ go run adv.go
```

# IDES

---

Ja hi ha IDEs  
amb suport per  
treballar dins de  
contenidors

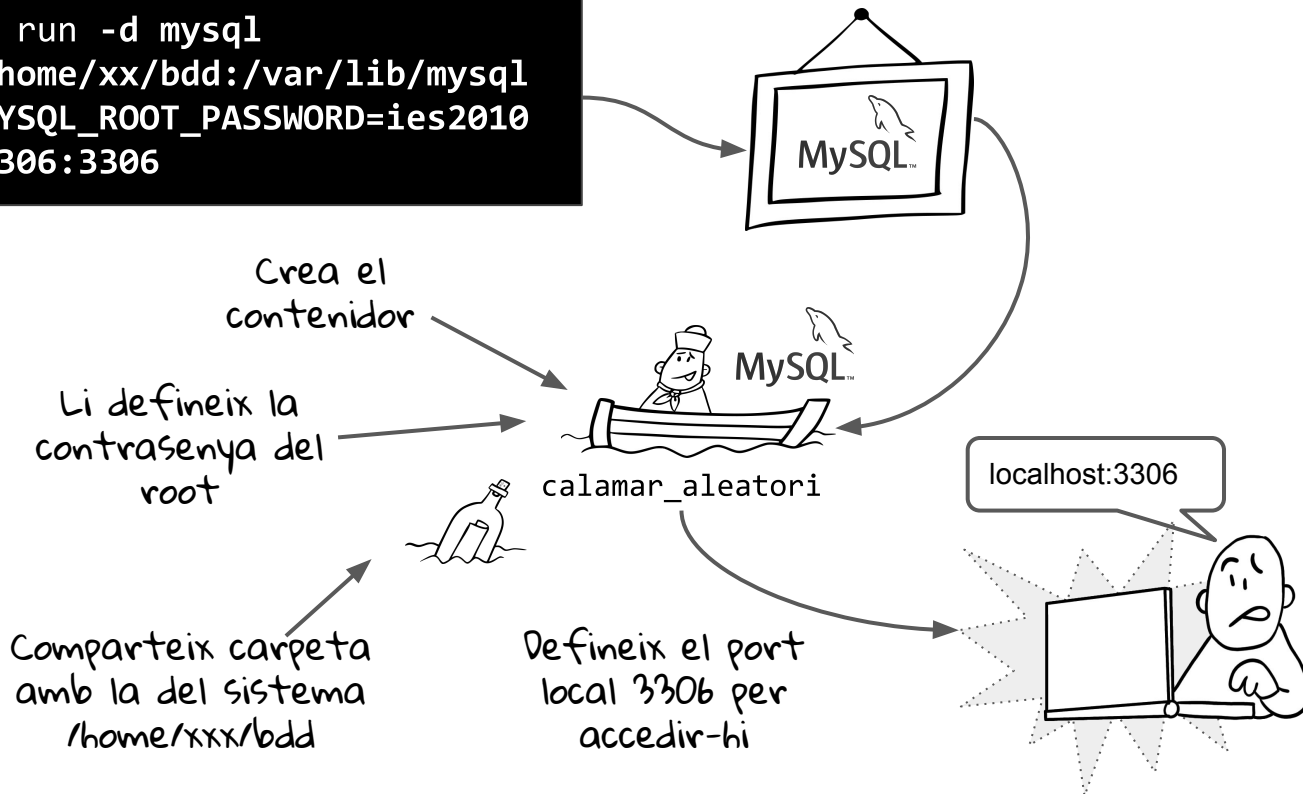




P5: Bases de dades

<https://github.com/utrescu/curs-docker-UdG/tree/day1/day1-5>

```
$ docker run -d mysql  
-v /home/xx/bdd:/var/lib/mysql  
-e MYSQL_ROOT_PASSWORD=ies2010  
-p 3306:3306
```





No en tinc prou!

# Més feina?

- Instal·lar Apache en un sistema Debian. Quina web surt per defecte?
- Compileu el programa en Rust que hi ha a <https://github.com/fraxiarsala/TasquesArbre.git> . Quin resultat dóna?
- Quants usuaris hi ha en la taula d'Usuaris de una imatge de SQLServer que es diu utrescur/testdata?





