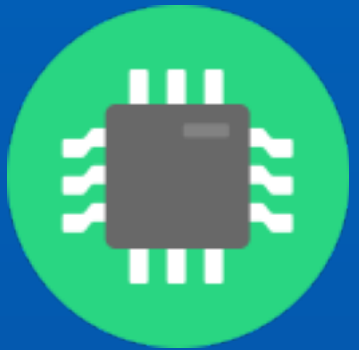


Usando Docker con sistemas Asterisk

Elio Rojano y Rosa Atienza

Advertencia



Esta presentación es puramente técnica.



Hay demostraciones que pueden salir mal.



Si ven comandos de consola, no se asusten.



Todos los ejemplos son hechos en consola.

Quiénes somos

Elio Rojano y Rosa Atienza

@hellc2 y *@atienzar*

Ingenieros Informáticos en Avanzada7

Editores y creadores de sinologic.net

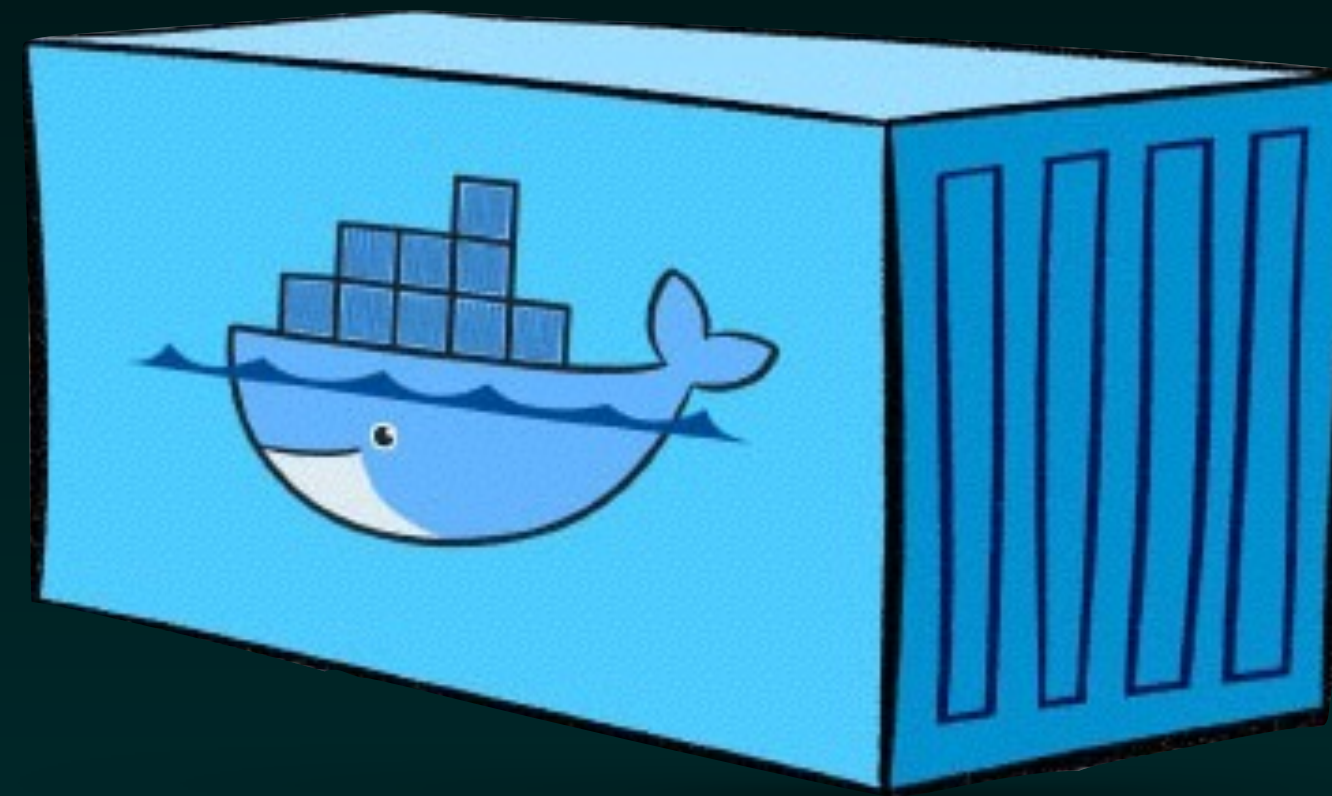
Trabajando desde 2004 con VoIP y Asterisk

Defensores del Software Libre y estándares abiertos



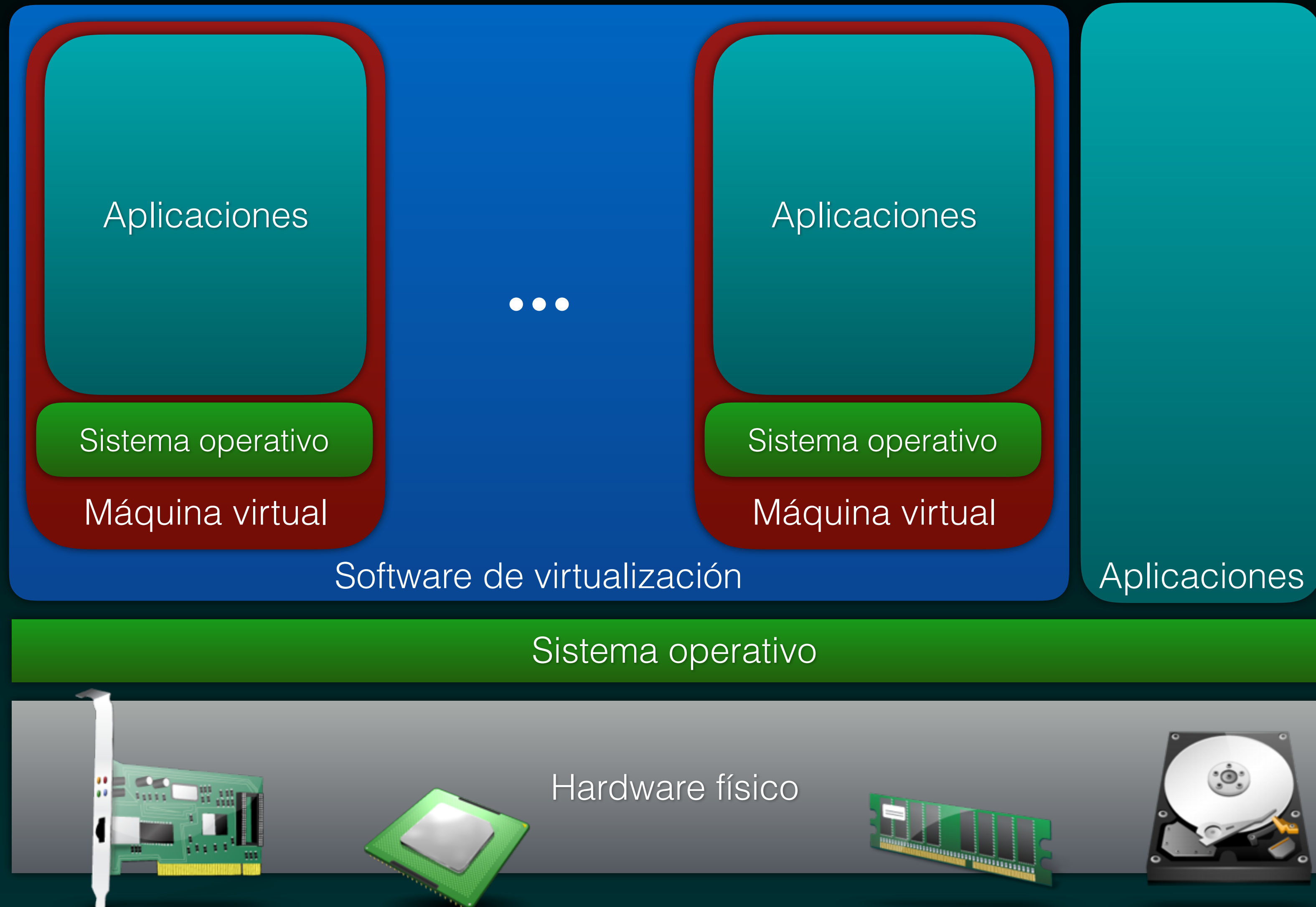
Introducción

¿Qué es un contenedor?

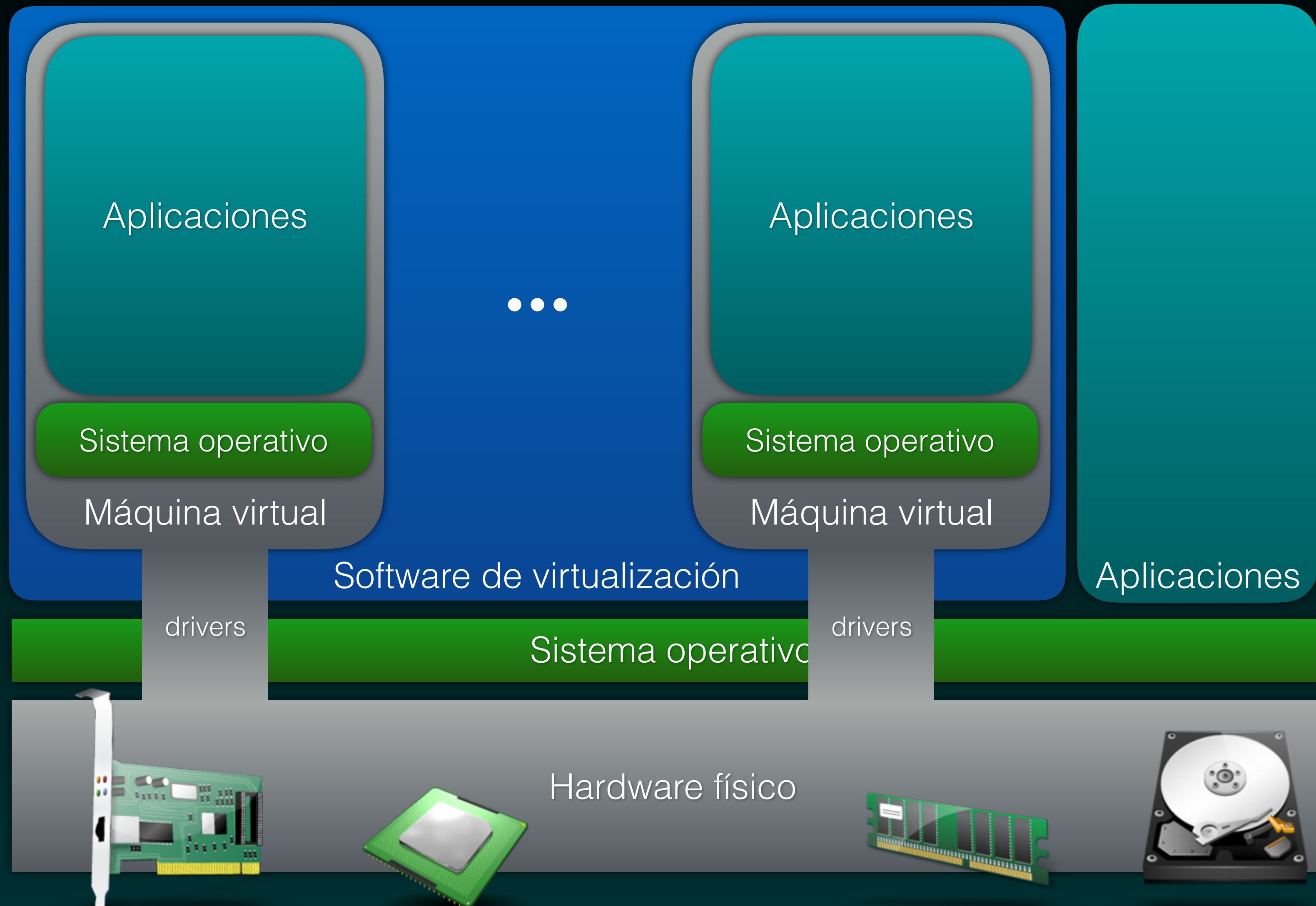


**El contenedor es
la unidad de medida
de la nube**

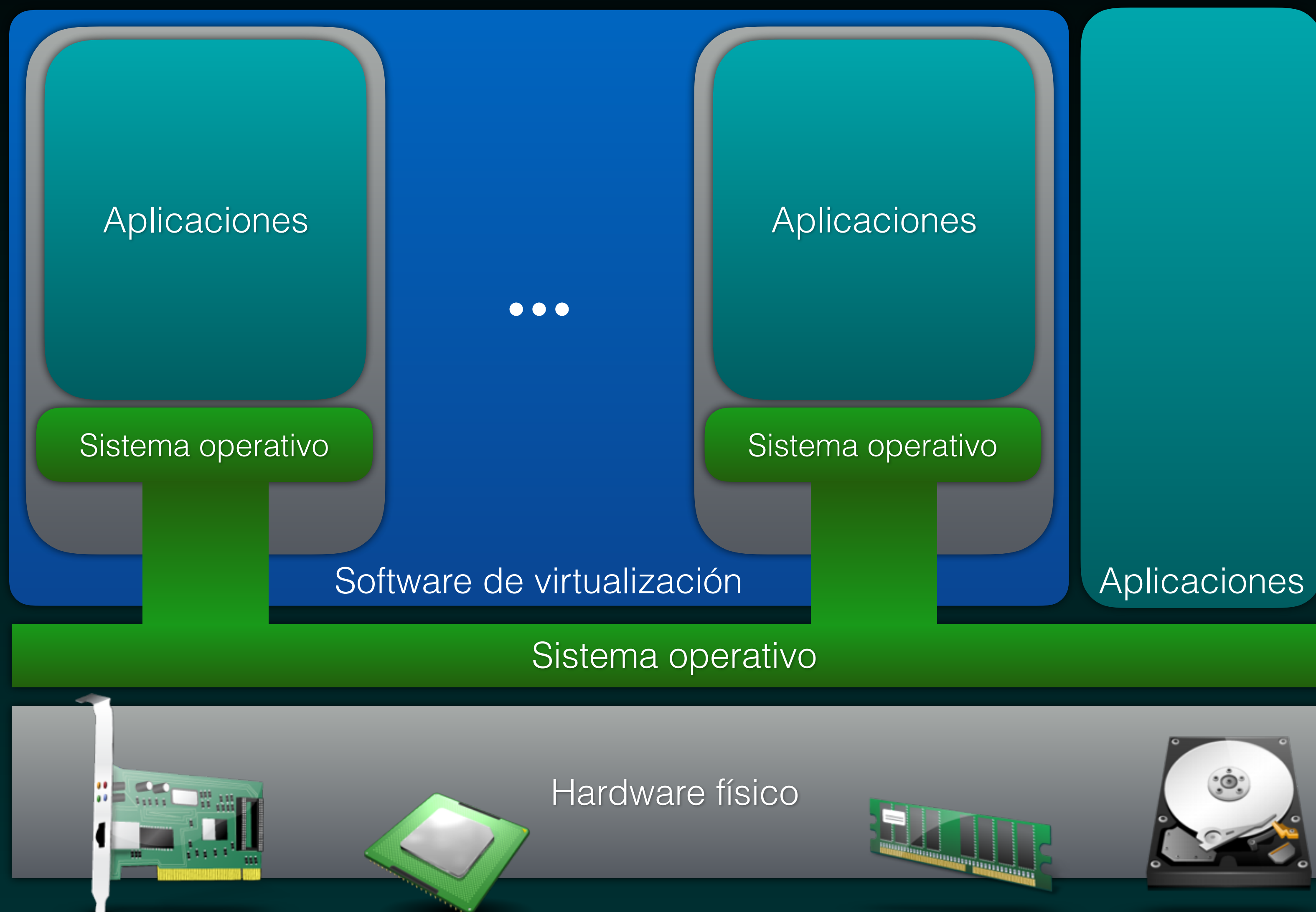
Virtualización



Paravirtualización



Contenedores

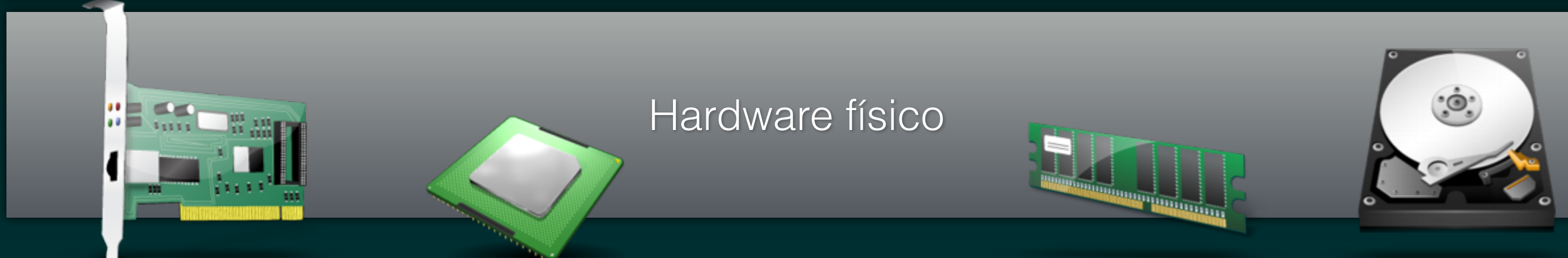


Contenedores

Contenedor 1

Contenedor 2

Contenedor 3





Docker es un Gestor de contenedores



Contenedores

Ubuntu

```
/bin  
/etc  
/home  
/lib  
/root  
/tmp  
/usr  
/var
```

Servidores

```
/etc/apache2  
/etc/mysql  
/etc/php5  
/usr/bin/apache*  
/usr/sbin/mysql*  
/usr/sbin/apache*  
/usr/lib/...  
/var/www/html
```

Wordpress

```
/var/www/wordpress  
/var/lib/mysql/tablas*
```

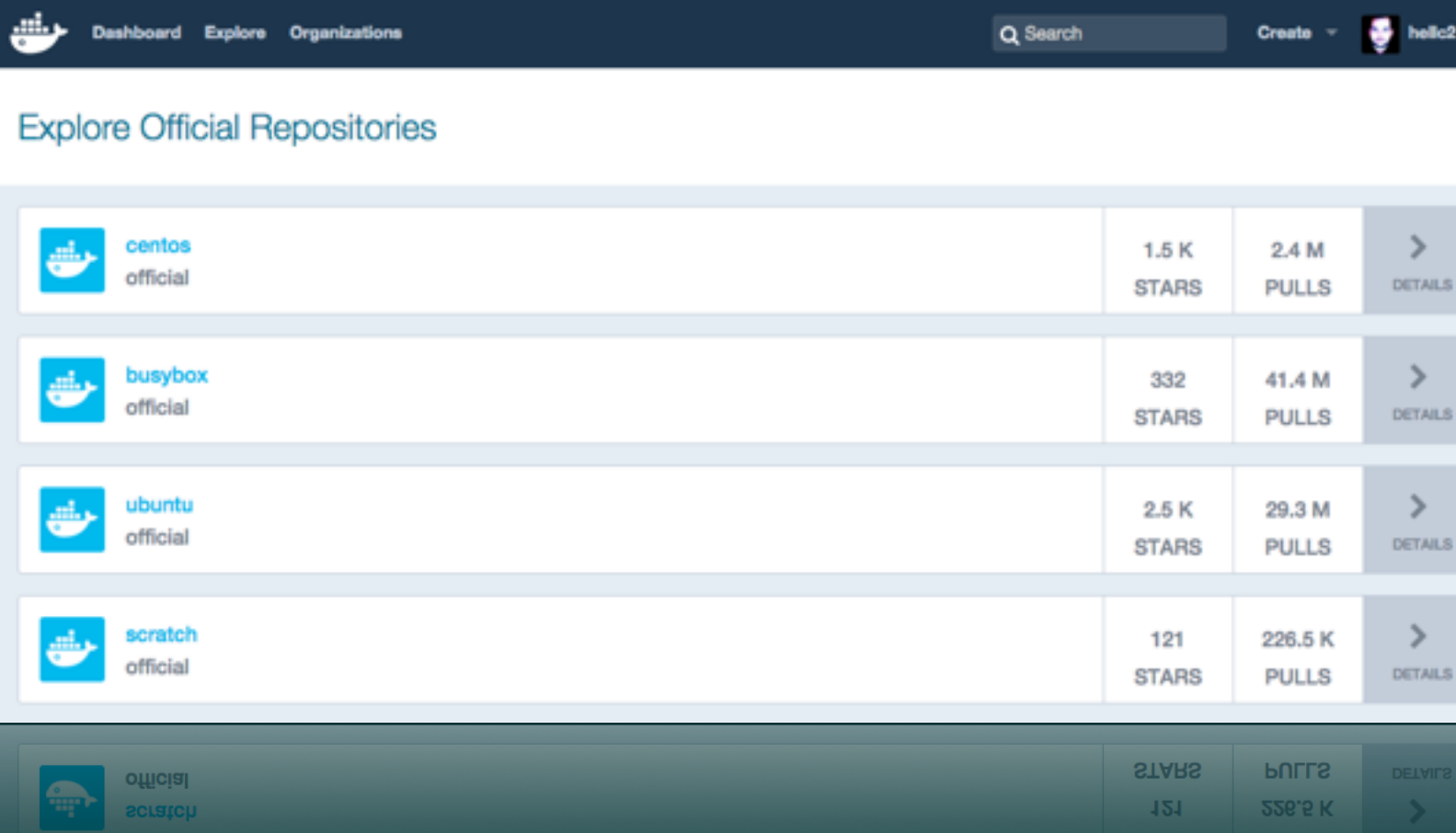
Sistema físico: **Kernel / Red / Disco duro / Memoria**

- Compatible con Linux, Windows y Mac
- Compatible con diferentes arquitecturas.
- Muy ligero y fácil de configurar.
- Intuitivo (una vez se conoce cómo funciona)
- Orientado a ser el “motor de la nube”
- Muchas utilidades compatibles.
- Herramienta ideal para *DevOps* (Developers + Sysadmins)
- Compatible con herramientas de clustering.
- 100% Software Libre

- Instalación sencilla en Debian:
`apt-get install docker.io`
- En entornos x86, oficialmente sólo es compatible en 64 bits.
- Incluye **Docker-Machine** para funcionar incluso en lugares no amistosos:
MacOSX, Windows, FreeBSD, etc...
- A pesar de todo, está programado en **Go**.



- Fácil de manejar gracias a sus repositorios **On-Line**.
- Tan flexible y potente como **GitHub** (DevOps ¿recuerdas?)
- Comandos similares: commit, push, pull, logs, info, etc.
- En lugar de **GitHub**, se utiliza hub.docker.com o docker.io
- Podemos buscar contenedores ya creados y utilizarlos.
- Los contenedores pasan a llamarse **usuario/nombre***
- *“Docker is doing to apt what apt did to tar”*



Repository	Stars	Pulls	Details
centos/official	1.5 K	2.4 M	>
busybox/official	332	41.4 M	>
ubuntu/official	2.5 K	29.3 M	>
scratch/official	121	226.5 K	>

Contenedores On-line

```
# docker search ubuntu
```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
ubuntu	Ubuntu is a Debian-based operating s...	2506	[OK]	
ubuntu-upstart	Upstart is an event-base lacement for ...	40	[OK]	
torusware/speedus-ubuntu	Always updated official Uu docker imag...	25		[OK]
sequenceiq/hadoop-ubuntu	An easy way to try HadoopUbuntu	23		[OK]
tleyden5iwx/ubuntu-cuda	Ubuntu 14.04 with CUDA drs pre-installed	18		[OK]
ubuntu-debootstrap	debootstrap --variant=min --components...	17	[OK]	
rastasheep/ubuntu-sshd	Dockerized SSH service, b on top of of...	14		[OK]
neurodebian	NeuroDebian provides neurence research...	13	[OK]	
guilhem/vagrant-ubuntu		11		[OK]
n3ziniuka5/ubuntu-oracle-jdk	Ubuntu with Oracle JDK. Ck tags for ver...	4		[OK]
sameersbn/ubuntu		4		[OK]
nimmis/ubuntu	This is a docker images derent LTS vers...	3		[OK]
nuagebec/ubuntu	Simple always updated Ubu docker images...	3		[OK]
ioft/armhf-ubuntu	[ABR] Ubuntu Docker imageor the ARMv7(a...	2		[OK]
armbuild/ubuntu-debootstrap	ARMHF port of ubuntu-debotrap	2		[OK]
maxexcloo/ubuntu	Docker base image built obuntu with Sup...	2		[OK]
isuper/base-ubuntu	This is just a small and an base Ubuntu...	1		[OK]
densuke/ubuntu-jp-remix	Ubuntu Linux	1		[OK]
seetheprogress/ubuntu	Ubuntu image provided by theprogress us...	1		[OK]
...				

Contenedores On-line

```
# docker run -it ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu

3fd0c2ae8ed2: Downloading 3.235 MB/65.67 MB
9e19ac89d27c: Download complete
ac65c371c3a5: Download complete
a5a467fddcb8: Download complete
library/ubuntu:latest: The image you are pulling has been verified. Important: image verification is
a tech preview feature and should not be relied on to provide security.

Digest: sha256:8b1bffa54d8a58395bae61ec32f1a70fc82a939e4a7179e6227eb79e4c3c56f6
Status: Downloaded newer image for ubuntu:latest
root@0db890674ded:/#
root@0db890674ded:/# cat /etc/apt/sources.list
deb http://archive.ubuntu.com/ubuntu/ trusty main restricted
deb-src http://archive.ubuntu.com/ubuntu/ trusty main restricted

root@0db890674ded:/#
```

Ubuntu bash



Docker

Descargar y ejecutar un comando del contenedor

`docker run contenedor comando`

Ejemplo:

```
# docker run centos ping -c4 www.google.com
```

```
PING www.google.com (216.58.210.132) 56(84) bytes of data.  
64 bytes from mad06s09-in-f132.1e100.net (216.58.210.132): icmp_seq=1 ttl=54 time=13.9 ms  
64 bytes from mad06s09-in-f4.1e100.net (216.58.210.132): icmp_seq=2 ttl=54 time=13.7 ms  
64 bytes from mad06s09-in-f4.1e100.net (216.58.210.132): icmp_seq=3 ttl=54 time=13.8 ms  
64 bytes from mad06s09-in-f132.1e100.net (216.58.210.132): icmp_seq=4 ttl=54 time=13.7 ms
```

```
--- www.google.com ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3248ms  
rtt min/avg/max/mdev = 13.737/13.831/13.947/0.143 ms
```

Una vez termine de ejecutar el comando, el contenedor vuelve a ser igual que recién descargada.

```
# docker run centos yum update
```

Esto actualizará el contenedor, pero al salir, volverá a tener los repositorios de paquetes iniciales.

Utilizaremos `-it` (interactive mode) para acceder al repositorio y poder ejecutar comandos.

Luego guardamos con `commit` los cambios.

Docker

```
# docker run -it centos bash
[root@2d71c2f762f5 /]# yum update
Loaded plugins: fastestmirror
base | 3.6 kB 00:00:00

...
=====
Package                Arch      Version              Repository            Size
=====
Updating:
coreutils              x86_64    8.22-12.el7_1.2      updates               3.2 M
krb5-libs              x86_64    1.12.2-15.el7_1      updates               826 k
[root@2d71c2f762f5 /]# exit
exit

root@debian:~# docker ps -a
CONTAINER ID        IMAGE               COMMAND              CREATED             STATUS              PORTS              NAMES
2d71c2f762f5       centos             "bash"              5 minutes ago      Exited
...

# docker commit kickass_saha centos
ce5bb110c28148a5350a406c94f0e42f2b012c2e2e6a8cd16db9a9de50a78014

# doker run -it centos bash
[root@e343f88993c1 /]# yum update
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
* base: centos.mirror.xtratelecom.es
* extras: centos.mirror.xtratelecom.es
* updates: centos.mirror.xtratelecom.es
No packages marked for update
```

Docker

Una manera de hacer contenedores es utilizando un archivo de texto llamado **Dockerfile**

```
FROM debian

RUN apt-get update
RUN apt-get -qqy install asterisk

EXPOSE 5060 10000 10001 10002 10003 10004

CMD ["asterisk", "-gvc"]
```

Dockerfile

```
# docker build --rm -t usuario/prueba .
```

```
Sending build context to Docker daemon 2.048 kB
```

```
Step 0 : FROM debian
```

```
----> d1f66aef36c9
```

```
Step 1 : RUN apt-get update
```

```
----> Running in 4a4355e387be
```

```
Get:1 http://security.debian.org jessie/updates InRelease [63.1 kB]
```

```
Ign http://httpredir.debian.org jessie InRelease
```

```
Get:2 http://security.debian.org jessie/updates/main amd64 Packages [185 kB]
```

```
Get:3 http://httpredir.debian.org jessie-updates InRelease [135 kB]
```

```
Get:4 http://httpredir.debian.org jessie Release.gpg [2373 B]
```

```
Get:5 http://httpredir.debian.org jessie Release [148 kB]
```

```
Get:6 http://httpredir.debian.org jessie-updates/main amd64 Packages [3619
```

```
Get:7 http://httpredir.debian.org jessie/main amd64 Packages [9035 kB]
```

```
Fetchd 9572 kB in 12s (787 kB/s)
```

```
Reading package lists...
```

```
----> ec7dd44b697d
```

```
Removing intermediate container 4a4355e387be
```

```
Step 2 : RUN apt-get -qqy install asterisk
```

```
----> Running in d239a7b5c231
```

Asterisk en un contenedor

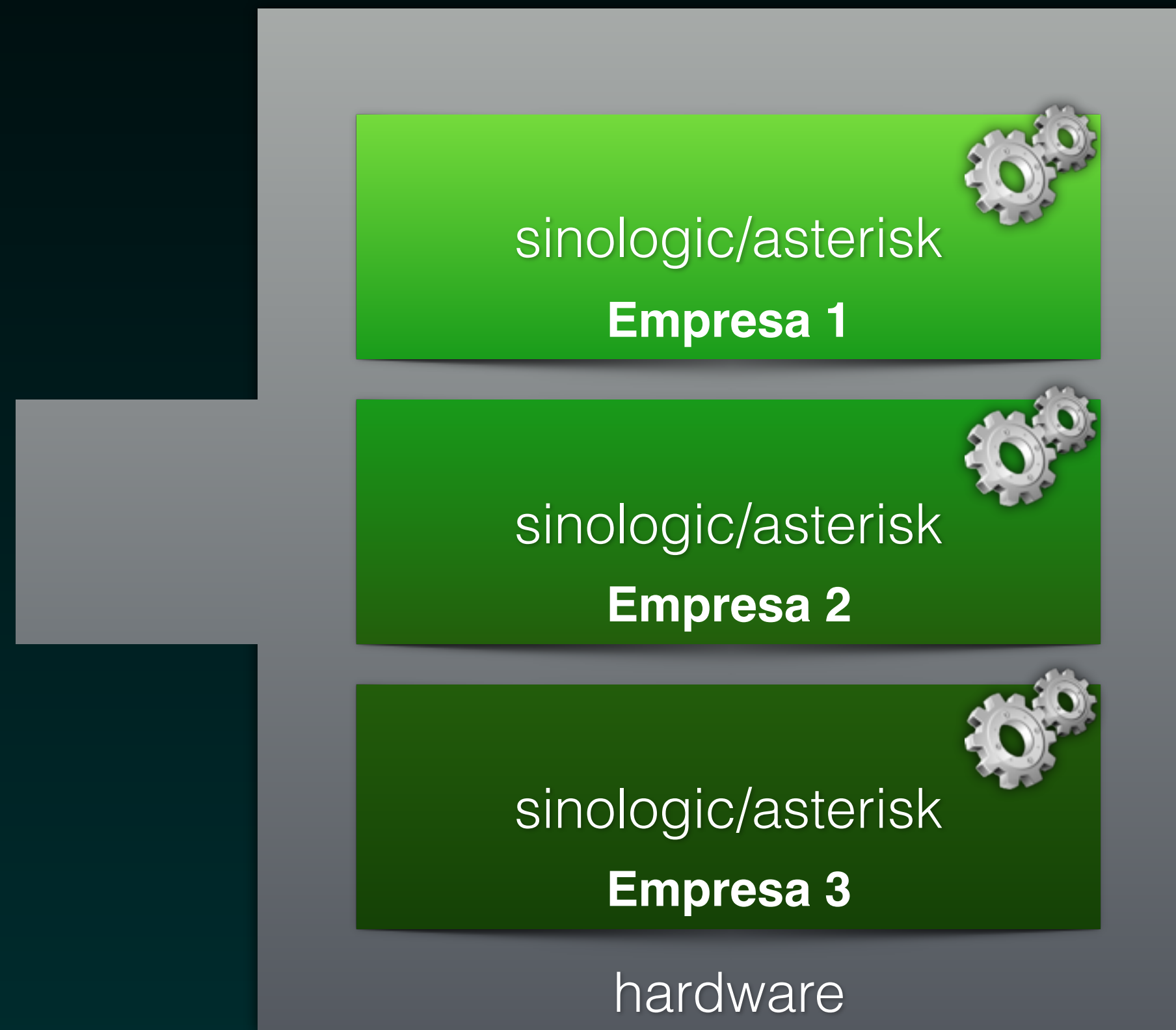


Asterisk en un contenedor

```
# docker run -it sinologic/asterisk asterisk -gvvvvvcdddddd
Parsing /etc/asterisk/asterisk.conf
Seeding global EID '02:42:ac:11:00:02' from 'eth0' using 'siocgifhwaddr'
Privilege escalation protection disabled!
See https://wiki.asterisk.org/wiki/x/1gKfAQ for more details.
Asterisk 11.13.1~dfsg-2+b1, Copyright (C) 1999 - 2013 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
[ Initializing Custom Configuration Options ]
Parsing /etc/asterisk/extconfig.conf
  == Parsing '/etc/asterisk/extconfig.conf': Found
Resetting translation matrix
Parsing /etc/asterisk/logger.conf
  == Parsing '/etc/asterisk/logger.conf': Found
  == Parsing '/etc/asterisk/asterisk.conf': Found
  == Manager registered action DBGet
...
  == Registered custom function 'QUEUE_MEMBER_PENALTY'
  app_queue.so => (True Call Queueing)
  == Parsing '/etc/asterisk/cli_permissions.conf': Found
Asterisk Ready.
  == Parsing '/etc/asterisk/cli.conf': Found
*CLI> _
```

Asterisk en un contenedor

Escalabilidad

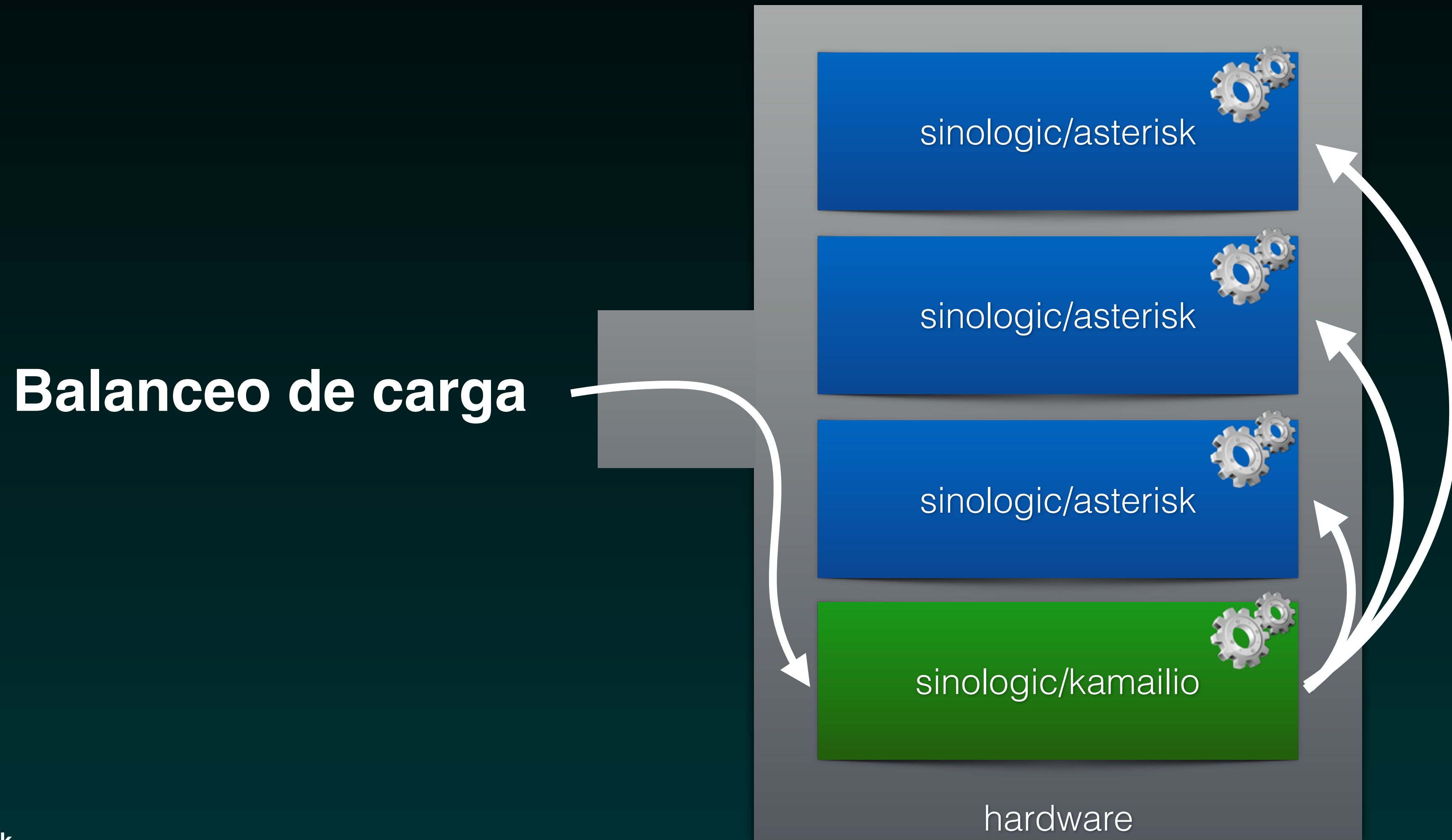


Asterisk en un contenedor

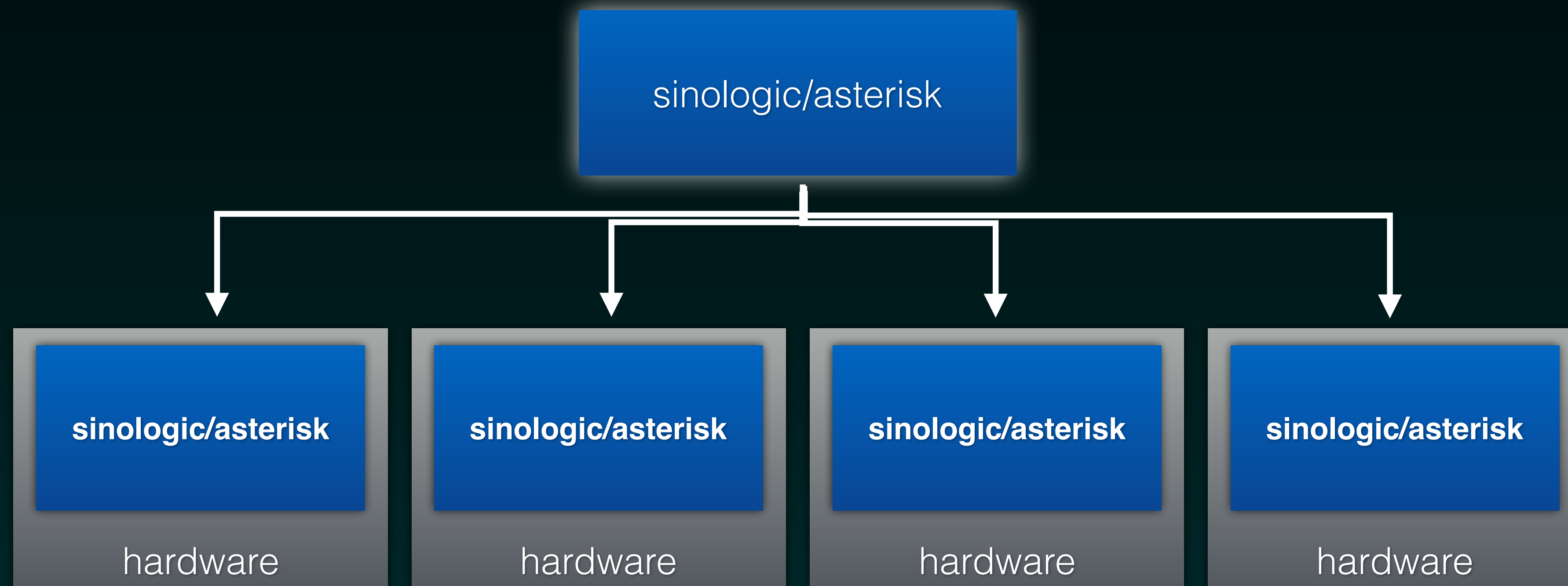
Alta disponibilidad



Asterisk en un contenedor



Asterisk en un contenedor



**Desarrollo de versiones
coordinadas**

Asterisk en un contenedor

Problemas

- El contenedor tiene una IP interna.
- Necesitamos mapear puertos.
- Los contenedores no tienen memoria...



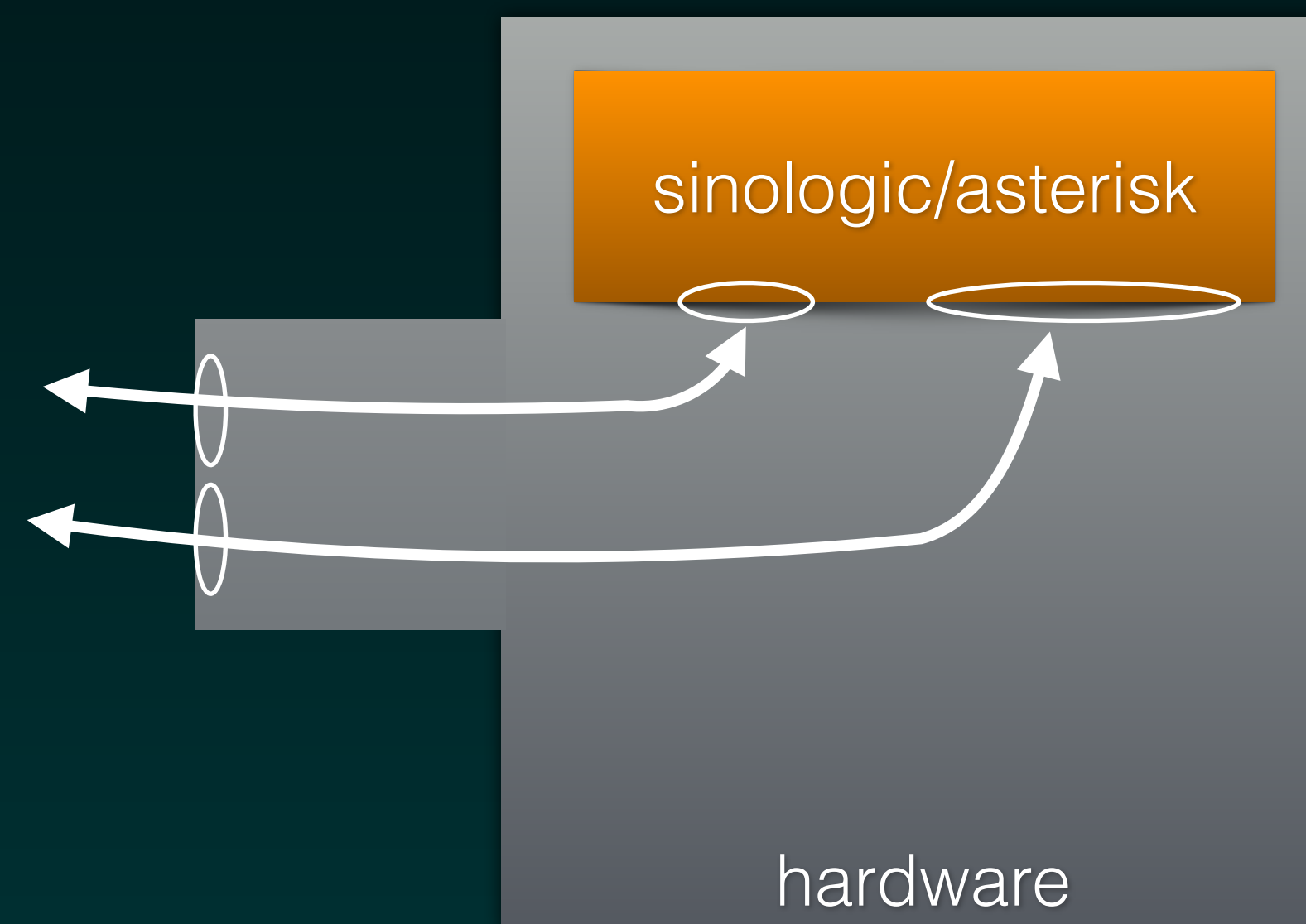
Todo tiene solución...

Asterisk en un contenedor

Cómo mapear puertos para utilizarlos en Asterisk

```
docker run -it -net=host  
  -p 5060:5060/udp  
  -p 10000-20000:10000-20000/udp  
  sinologic/asterisk asterisk -gvvvvvc
```

Cuidado!, el *mapeo de puertos* consume mucho procesador!
En caso de varios contenedores, los puertos no deben coincidir.



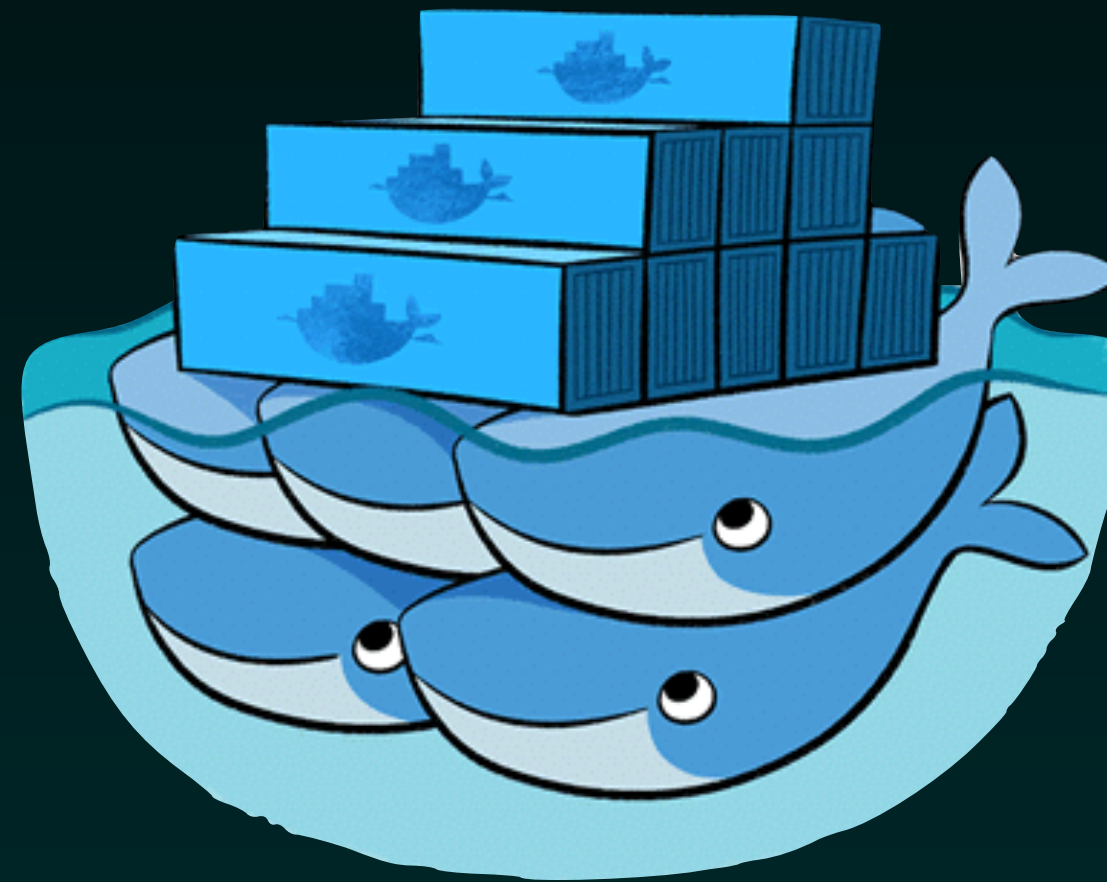
Asterisk en un contenedor

Si queremos personalizar una configuración, debemos “mapear un directorio”

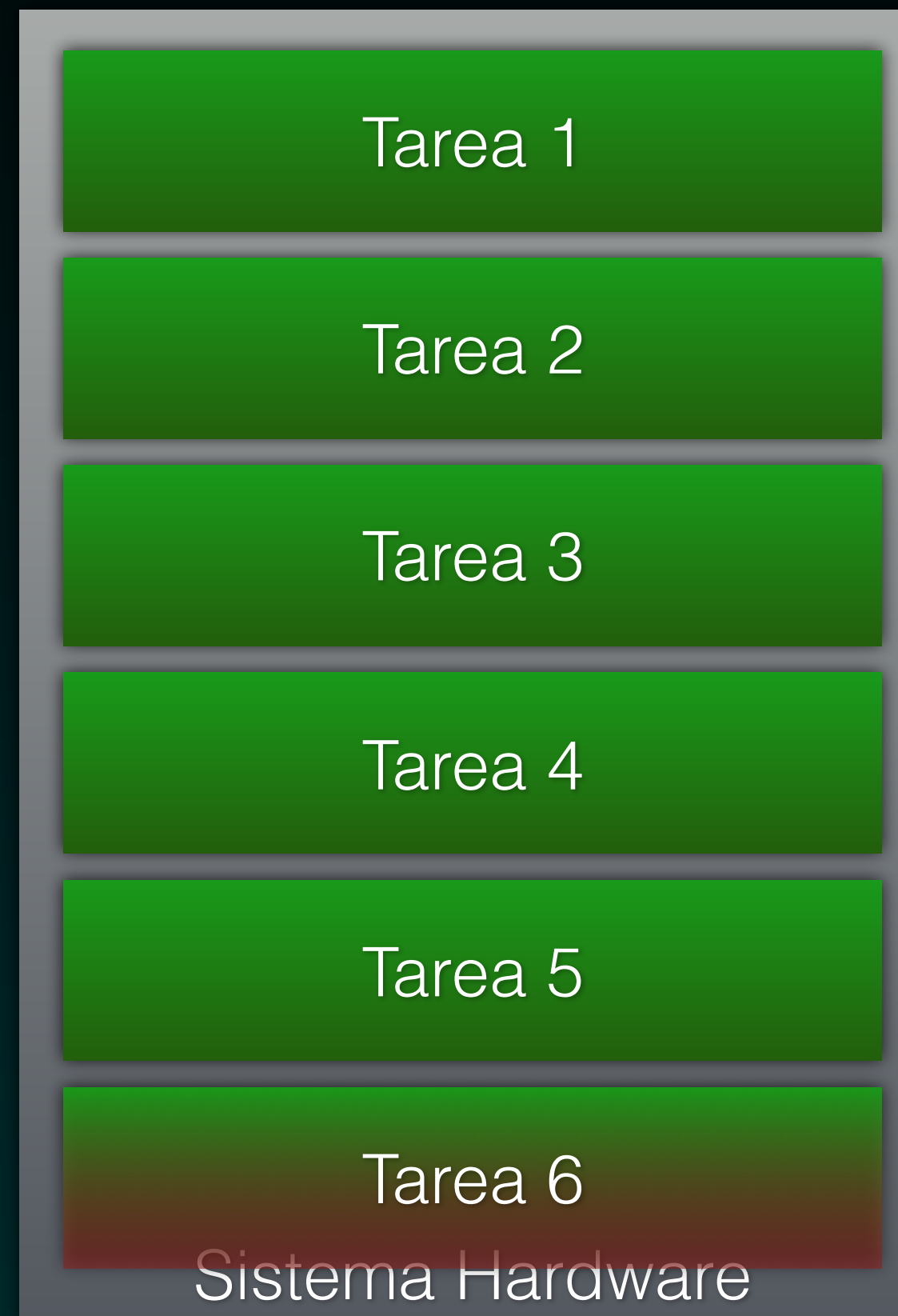
```
docker run -it -net=host
  -p 5060:5060/udp
  -p 10000-20000:10000-20000/udp
  -v /etc/asterisk:/etc/asterisk
  sinologic/asterisk asterisk -gvvvvvvc
```



Cluster de contenedores



Qué es un Cluster



Qué es un Cluster



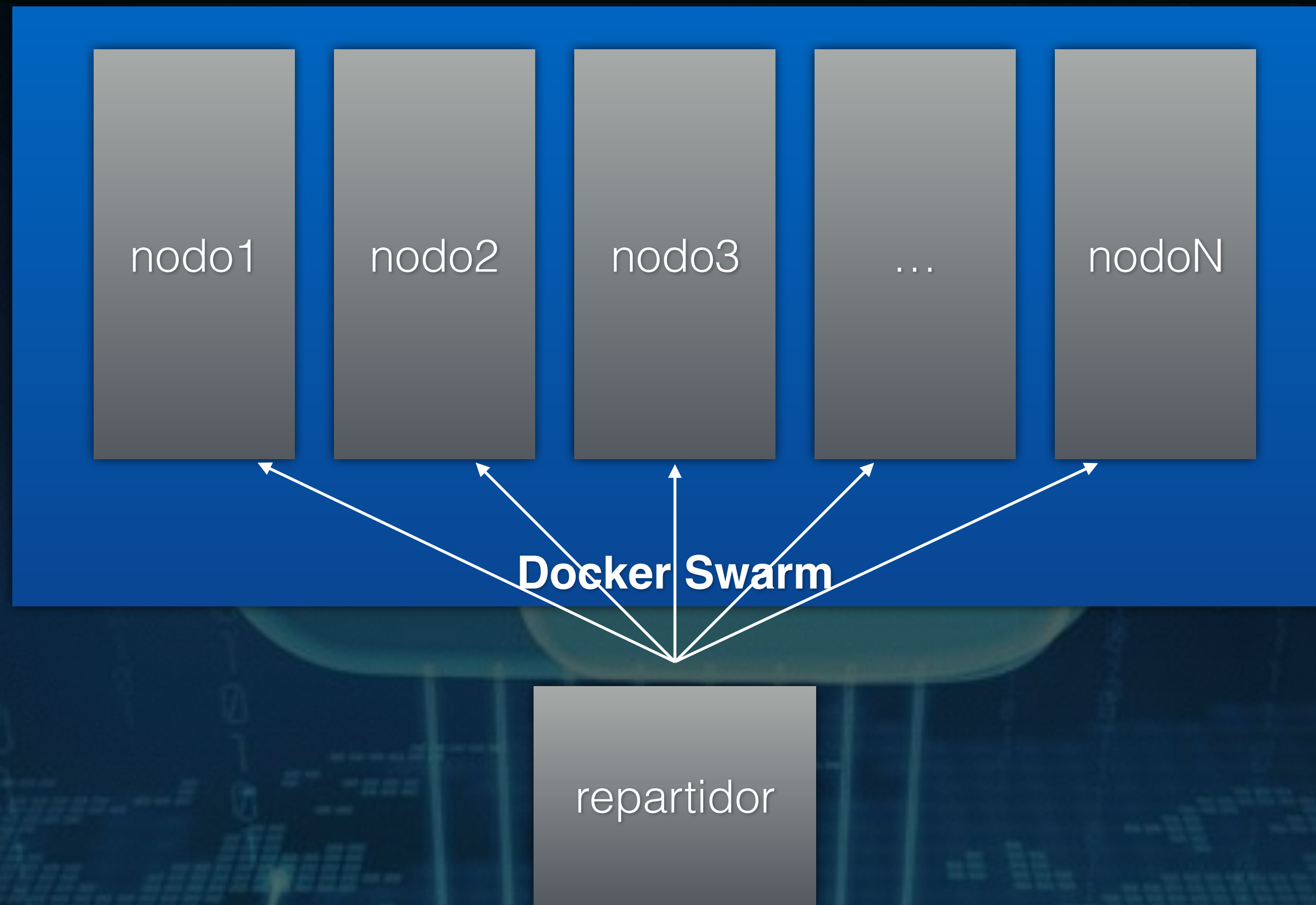
Qué es un Cluster



tareaN, ... , tarea3, tarea2, tarea1

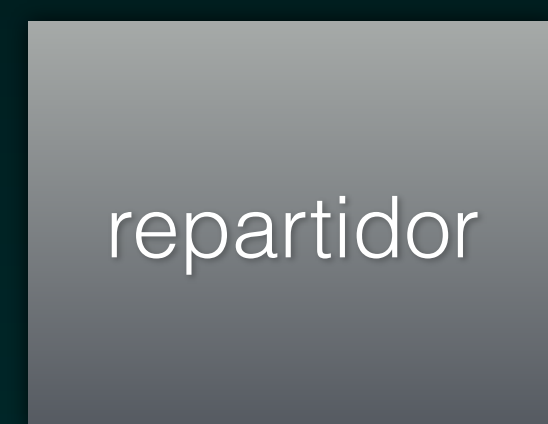
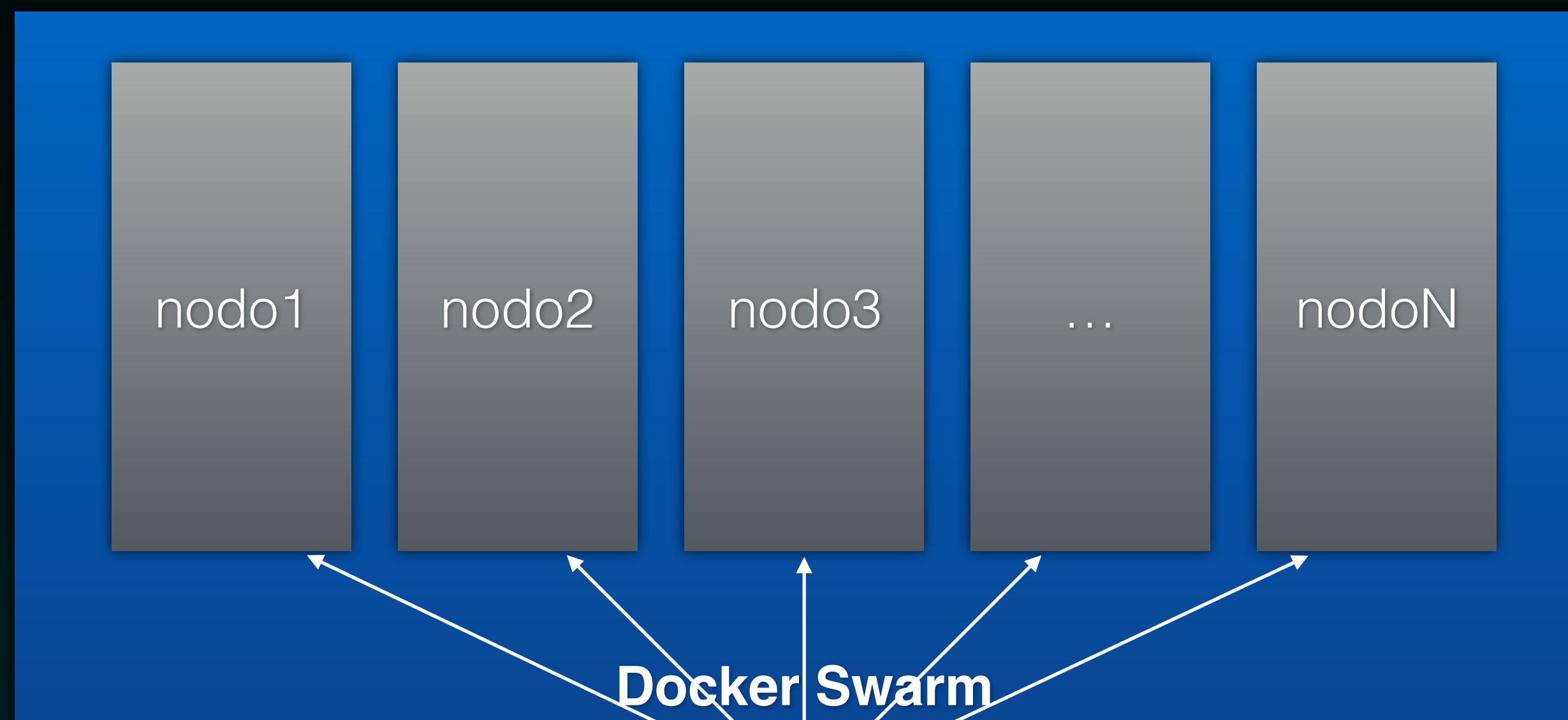


Qué es un Cluster



tareaN, ... , tarea3, tarea2, tarea1 

Qué es un Cluster

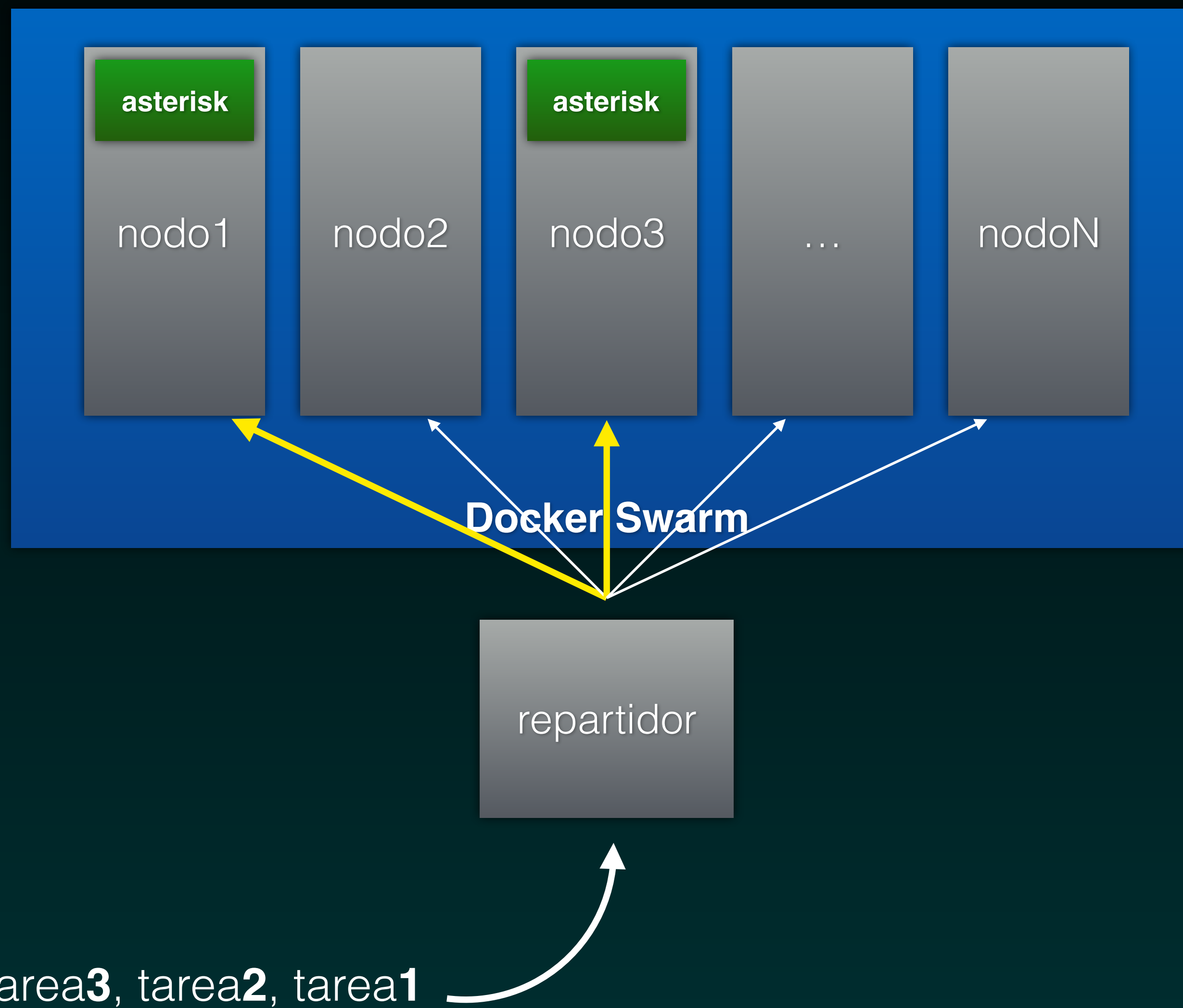


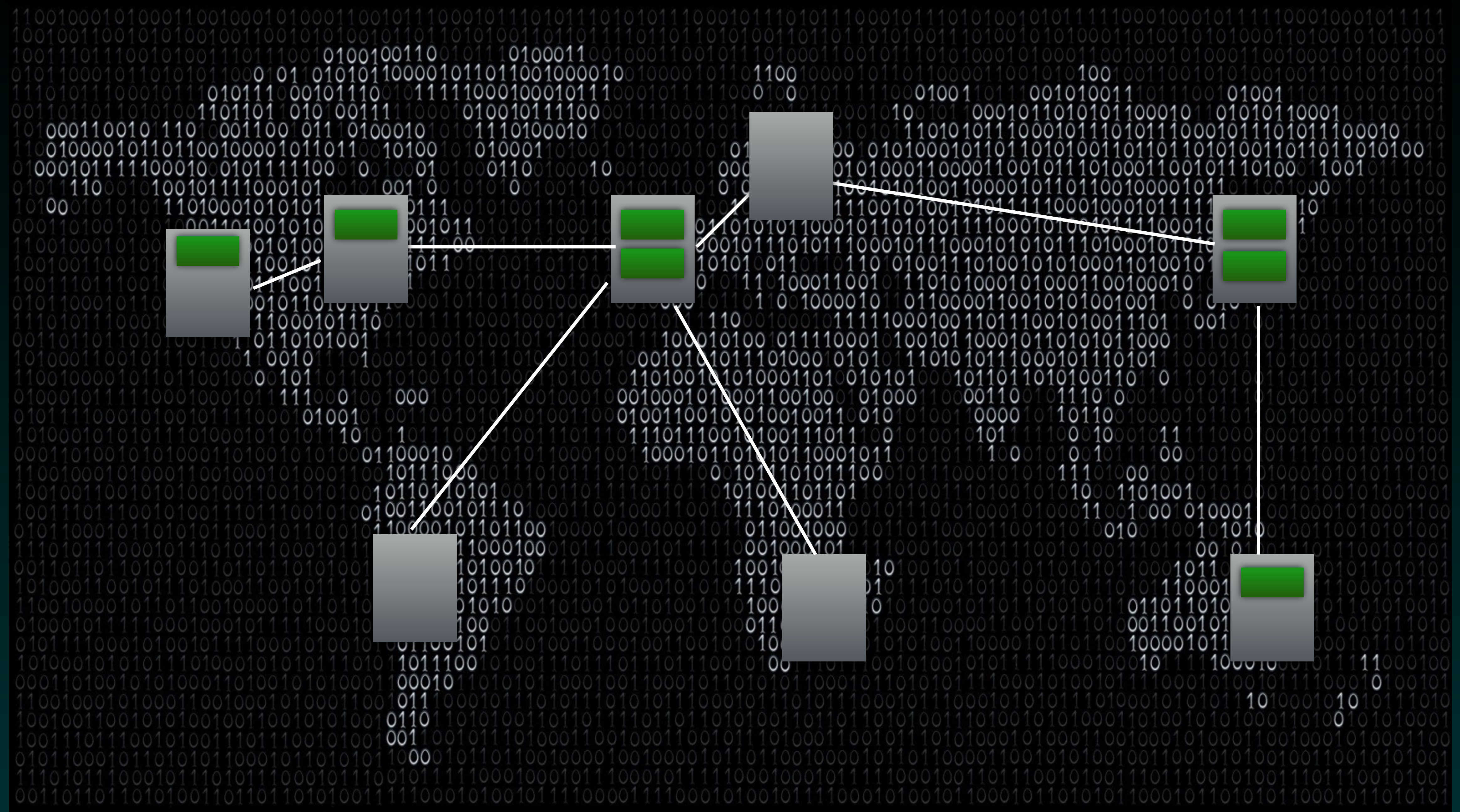
```
docker run -d sinologic/asterisk asterisk -gvc  
docker run -d sinologic/asterisk asterisk -gvc
```

tareaN, ... , tarea3, tarea2, tarea1

*Cada **Asterisk** actúa como un proceso distribuido

Qué es un Cluster



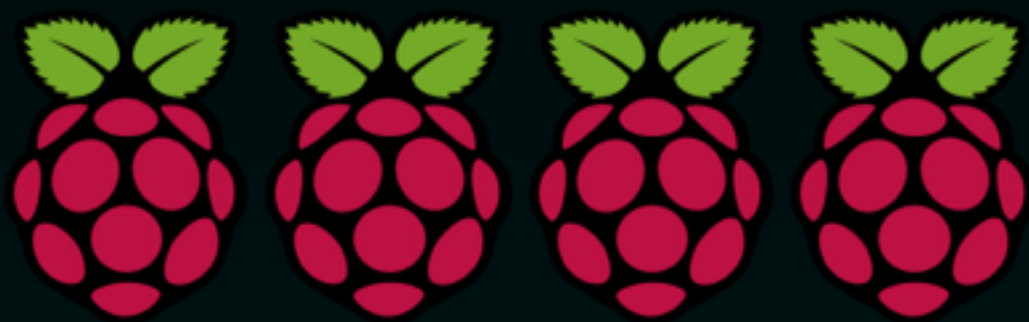


DEMO

Vamos a hacerlo en pequeño



Presentación del Nodo



Sistema: Raspberrry PI 2 Model B v.1.1

Procesador: ARM v.7 con 4 núcleos

Memoria RAM: 1Gb

SO: Raspbian Jessie September 2015

Kernel: 4.1

Docker versión: hypriot 1.8.3





Nodo 1




Nodo 2

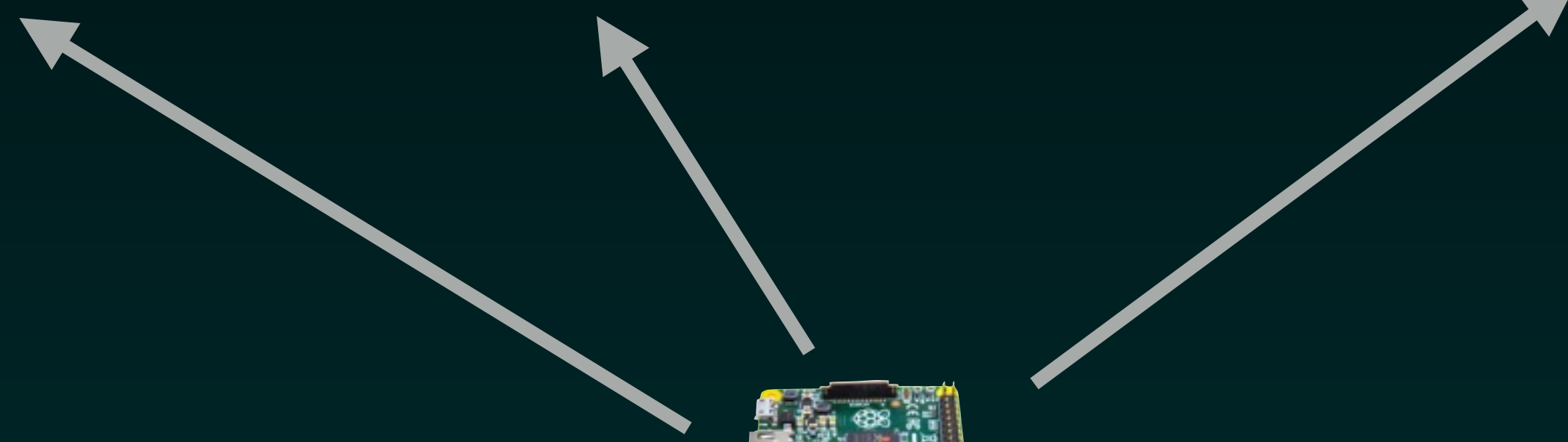


...

Nodo N




comandos



¿PREGUNTAS?

Gracias!

- A **los creadores** de **Docker** y **Asterisk** por crear estas grandes aplicaciones.
- A **Avanzada7** por invitarnos a dar estas conferencias.
- A **Pablo** por dejarnos algo de tiempo para investigar y hacer esta charla.
- A **los lectores** de sinologic.net.
- A **vosotros** por seguir aquí.