

T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es

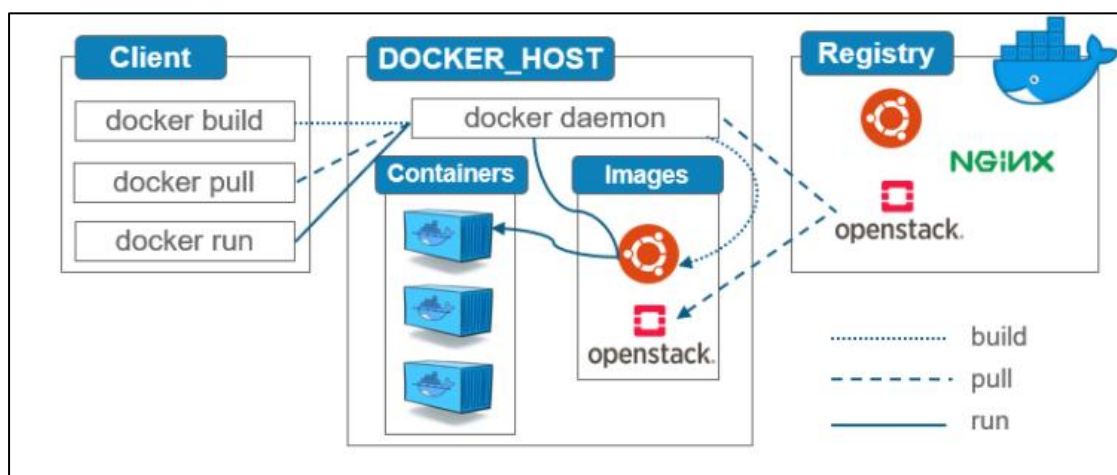
3. Docker. Arquitectura, instalaci n, administraci n

1.	Arquitectura de Docker	2
2.	Docker Engine.	4
3.	Productos Docker.	5
4.	Instalaci�n de Docker Desktop.....	5
4.1.	Instalaci�n de Docker Desktop para Windows.	6
1)	Requisitos del sistema. Tareas previas.....	6
4.2.	Instalaci�n Docker Desktop para otras plataformas.	8
5.	Administraci�n b�sica	9

T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es

1. Arquitectura de Docker

Docker utiliza una **arquitectura** basada en el modelo **cliente-servidor**. El **Docker client (cliente)** hace las peticiones de servicio y se comunica con el **Docker daemon (servidor)**, que hace el trabajo de compilar, ejecutar y distribuir los contenedores. El client y el daemon pueden ejecutarse en la misma m quina, o bien un client se puede conectar a un daemon remoto. El client y el daemon de Docker se comunican mediante una **API**.



Componentes principales de un despliegue o instalaci n de Docker:

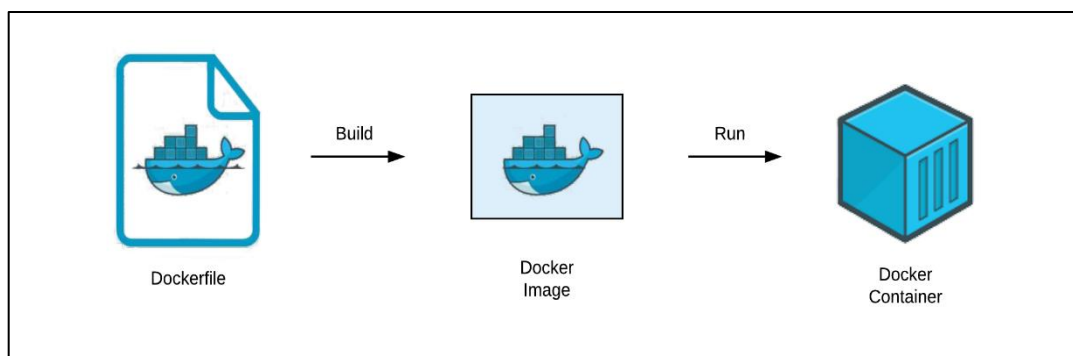
- **Docker daemon:** un daemon es una aplicaci n o **servicio residente en memoria**, **no interactivo**, que **se ejecuta en segundo plano de forma persistente**. El daemon de Docker **escucha** las **solicitudes** de la API de Docker y **administra objetos** de Docker, como im genes, contenedores, redes y vol menes. Un daemon tambi n puede comunicarse con otros daemons para administrar los servicios de Docker.
- **Docker client:** un client tambi n es una **aplicaci n**, en este caso **interactiva** con el usuario. Es la forma principal en que muchos usuarios de Docker interact an con la herramienta. Cuando se utilizan comandos, como por ejemplo "docker run..." (para ejecutar o iniciar un contenedor), el client env a este comando al daemon, que lo ejecuta. El Docker client puede comunicarse con m s de un daemon.

T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es

- **Docker API:** interfaz para las comunicaciones entre Docker client y Docker daemon.
- **Docker registry:** almacena im genes de Docker. **Docker Hub** es un **registro p blico** que cualquiera puede usar. Docker est  configurado para buscar im genes en Docker Hub de forma predeterminada. Se pueden utilizar im genes del registro p blico para iniciar contenedores, tambi n se puede enviar im genes propias al registro p blico.
- **Objetos Docker:** durante la utilizaci n de Docker, se crean, gestionan y utilizan una serie de objetos. A continuaci n, detallamos los dos objetos m s relevantes:
 - **Im genes:** cada imagen es una **plantilla de solo lectura con las instrucciones necesarias para crear un contenedor Docker**. Es como un molde para iniciar contenedores. Es muy habitual que una imagen se base en otra imagen, con alguna personalizaci n adicional. Por ejemplo, se puede crear una imagen basada en la imagen oficial de Ubuntu, pero instalando adicionalmente un servidor web Apache y una aplicaci n web propia, as  como los detalles de configuraci n necesarios para ejecutar dicha aplicaci n. Pudiendo crear as  nuestras propias im genes o pudiendo usar las creadas por otros y que est n publicadas en un registro o repositorio p blico.
Para crear una imagen propia o personalizada, necesitamos crear un fichero **Dockerfile**, donde se definir n los **pasos necesarios para crear la imagen**. Cada **instrucci n** en un fichero Dockerfile, crea **una capa** en la imagen. **Cada capa se va montando por orden, “encima” de las anteriores, de forma incremental.**
 - **Contenedores:** un contenedor es una **instancia ejecutable de una imagen**. Un contenedor, se puede crear, iniciar, detener, mover o eliminar, mediante la API de Docker. Se puede conectar un contenedor a una o m s redes, adjuntarle almacenamiento o incluso crear una nueva imagen basada en su estado actual. **Un contenedor se define por su imagen, as  como por las opciones de configuraci n que se le proporcione cuando se crea o se inicia el**

T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es

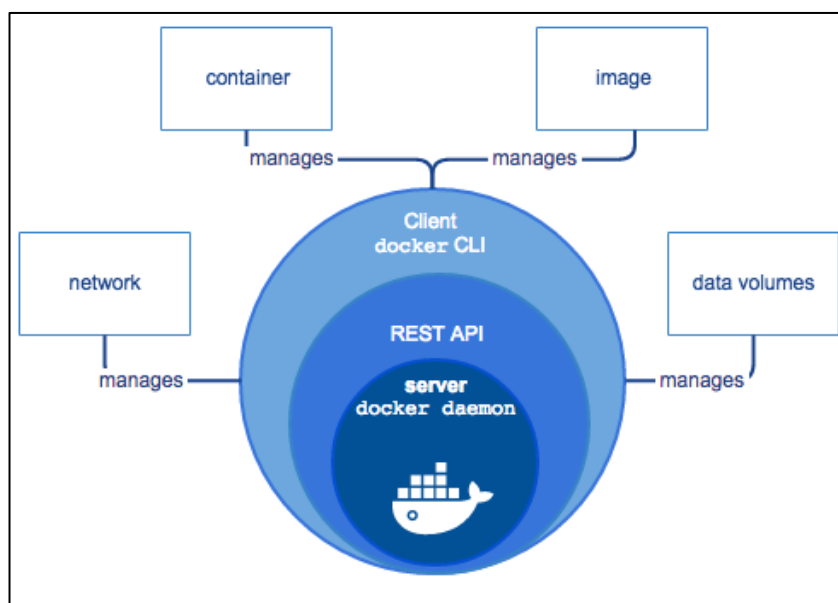
contenedor. Cuando se detiene un contenedor, cualquier cambio en su estado que no est  en un almacenamiento persistente desaparece.



2. Docker Engine.

Docker Engine es la **parte central de toda instalaci n Docker**. Es un **servicio** que act a como una aplicaci n **cliente-servidor**, con los siguientes **3 componentes**:

- **Docker daemon:** un **servidor** con un proceso de tipo daemon, explicado anteriormente.
- **Docker CLI:** un **cliente** de interfaz de l nea de comandos (Command Line Interface). Utiliza las API para controlar o interactuar con el daemon mediante comandos.
- **Docker API:** una **interfaz** que posibilita la comunicaci n entre Docker CLI y Docker daemon.



T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es

3. Productos Docker.

Docker es un **proyecto de software libre y c digo abierto**, que se distribuye mediante diferentes **niveles de suscripci n**. Este formato consiste en diferentes tipos de licenciamiento, desde un nivel gratuito con ciertas limitaciones, hasta un nivel m s completo en cuanto a herramientas y soporte, orientado a organizaciones con cierto volumen dedicadas al desarrollo profesional de aplicaciones:

- **Personal:** este nivel de suscripci n es **gratuito**. Est  pensado para desarrolladores individuales, capacitaci n, comunidades de c digo abierto y peque as empresas.
- **Pro:** este nivel de suscripci n es de pago, aunque el de menor coste. Incluye herramientas profesionales para desarrolladores individuales que desean acelerar su productividad.
- **Team:** este nivel de suscripci n es de pago y con un coste m s alto que el anterior. Est  pensado para equipos de trabajo, incluye capacidades de colaboraci n, productividad y seguridad.
- **Business:** este nivel de suscripci n es de pago, el de mayor coste de todos. Pensado para medianas y grandes empresas que necesitan administraci n centralizada y capacidades de seguridad avanzadas.

4. Instalaci n de Docker Desktop.

A t tulo informativo, **la instalaci n de Docker Desktop incluye:** Docker Engine , Docker Compose, Kubernetes, etc. <https://www.docker.com/products/docker-desktop/>



T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es

4.1. Instalaci n de Docker Desktop para Windows.

Para llevar a cabo la instalaci n de Docker Desktop para Windows, podemos acceder al **sitio web oficial de Docker**, <https://docs.docker.com/desktop/windows/install/>

En este sitio web encontraremos tanto el **software** necesario, como los **manuales e instrucciones** detalladas de c mo proceder con la instalaci n en cada caso. A continuaci n, se detallan ciertas particularidades importantes a tener en cuenta:

1) Requisitos del sistema. Tareas previas.

Para poder llevar a cabo la instalaci n de Docker Desktop para Windows, hay que cumplir una serie de requisitos previos. Si con tu PC no los pudieras cumplir, no pasa nada, no hay que comprar uno nuevo, contacta con el profesor para buscar una alternativa.

Se recomienda lanzar el instalador y que sea  l quien revise el sistema y nos indique lo que es necesario realizar, si fuera necesario realizar alguna acci n de forma previa.

Por un lado, nos encontraremos los t picos requisitos a nivel de hardware y versi n de sistema operativo. No son demasiados, ni muy estrictos, en un alto porcentaje de casos se cumplir n.

Por otro lado, puede que sea necesario realizar alguna de las siguientes tareas previas:

- **Disponer de WSL2 (Windows Subsystem for Linux) actualizado:** WSL es una capa de compatibilidad desarrollada por Microsoft para poder usar herramientas y comandos propios de Linux, en Windows. **Lo m s habitual es que el sistema ya disponga de esta capa, puede que  nicamente requiera una actualizaci n.**

Si ejecutas el instalador y te pide actualizar WSL, puedes descargar el paquete de actualizaci n para instalarlo desde aqu :

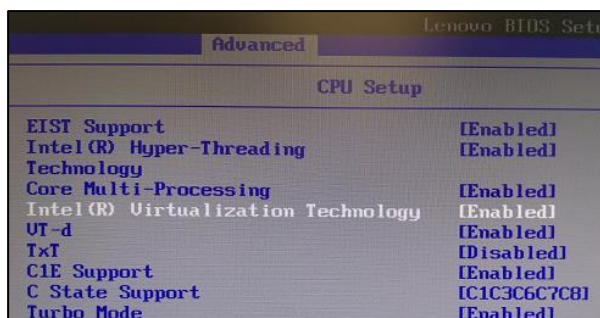
<https://docs.microsoft.com/es-es/windows/wsl/install-manual#step-4---download-the-linux-kernel-update-package>

T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es

Si necesitaras instalar WSL desde cero, lo cual es poco probable, puedes seguir las siguientes indicaciones:

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

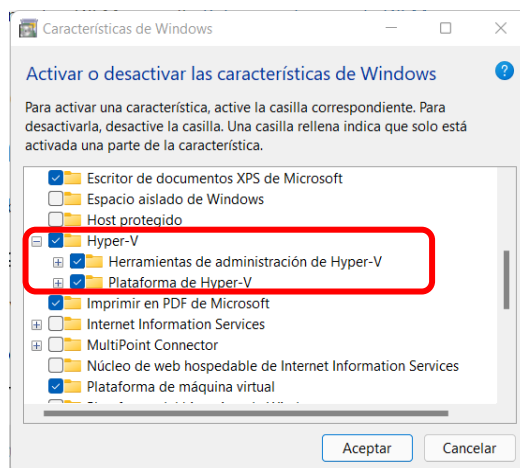
- La tecnolog a de virtualizaci n debe estar habilitada a nivel de hardware (procesador). Si nos lo pide el instalador, para habilitar la tecnolog a de virtualizaci n a nivel de procesador, ser  necesario acceder a la BIOS (Sistema de entrada/salida b sico) de nuestra m quina o PC, buscar en la **configuraci n de CPU** el campo que hace referencia a la virtualizaci n y **habilitarlo** ("enabled"), guardar los cambios en la configuraci n y salir.



Cuando se reinicie la m quina, lo har  con el nuevo cambio de configuraci n. Hay que tener en cuenta que, en funci n de la marca y modelo de nuestro PC debemos averiguar antes, por ejemplo, mediante una consulta en Internet, c mo acceder a la BIOS de nuestro equipo. Esto es debido a que dependiendo del fabricante puede hacerse mediante diferentes procedimientos. Una vez dentro, debemos localizar el campo de la configuraci n a habilitar, la denominaci n del campo podr a variar. En la imagen puedes ver un ejemplo orientativo.

Título	3. Docker. Arquitectura, instalación, administración
Destinatario	1º DAW - Semipresencial
Autor	Pascual Martínez
Correo	pmartinez@florida-uni.es

Si nos lo pidiera el instalador, podemos **habilitar la tecnología de virtualización a nivel de sistema operativo en Windows**, desde las **Características de Windows**.



Una vez finalizados los requisitos previos, podemos seguir el proceso de instalación mediante el asistente de Docker Desktop para Windows.

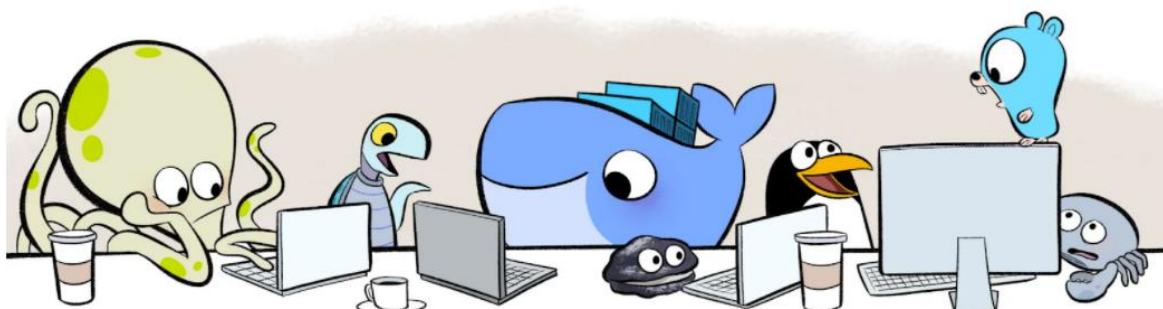
4.2. Instalación Docker Desktop para otras plataformas.

La instalación de Docker Desktop para otras plataformas no suele requerir las tareas previas descritas anteriormente para Windows.

La instalación de **Docker Desktop para Mac**, viene descrita en el sitio web oficial de Docker: <https://docs.docker.com/desktop/mac/install/>

La instalación de **Docker Desktop para Linux**, está soportada para las distribuciones Ubuntu, Debian y Fedora. Viene descrita en sitio web oficial de Docker:

<https://docs.docker.com/desktop/install/linux-install/>



T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es

5. Administraci n b sica

Una vez instalado Docker Desktop podemos acceder a la consola, terminal, shell, bash, s mbolo de sistema o powershell, en el caso de Windows.

Es decir, a la **l nea de comandos (CLI)** del sistema operativo.

Si escribimos **“docker version”**, nos debe devolver **la versi n tanto del cliente como del servidor instalados**.

```

Windows PowerShell
PS C:\Windows\System32> docker version
Client:
 Cloud integration: v1.0.35+desktop.11
 Version: 25.0.3
 API version: 1.44
 Go version: go1.21.6
 Git commit: 4debf41
 Built: Tue Feb 6 21:13:02 2024
 OS/Arch: windows/amd64
 Context: default

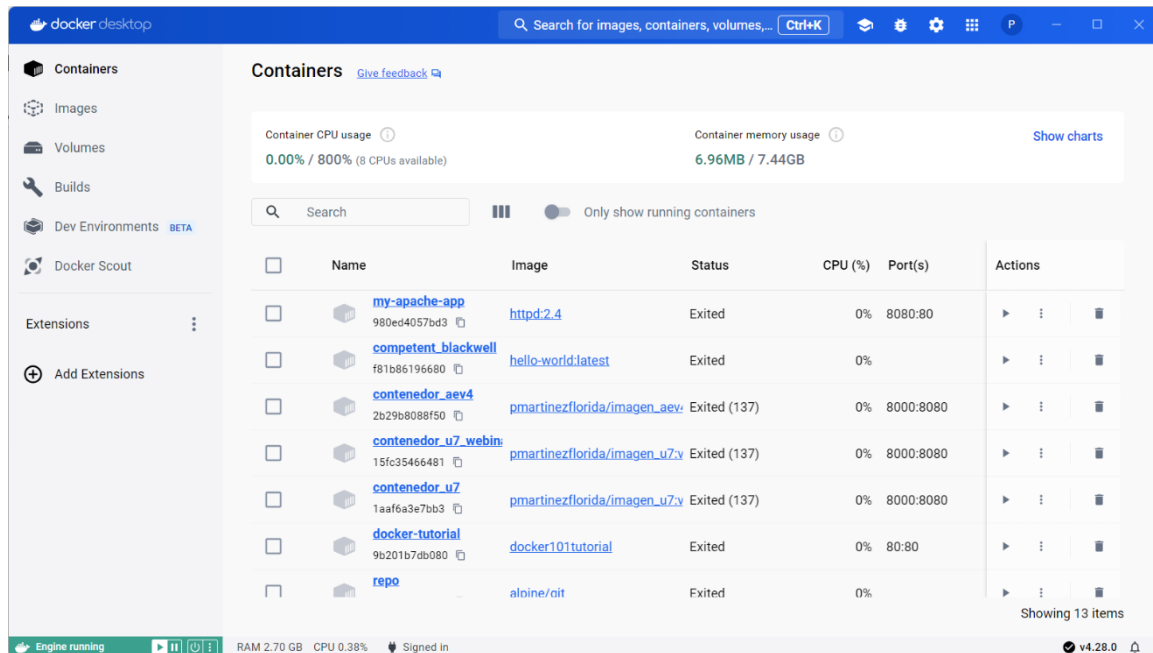
Server: Docker Desktop 4.28.0 (139021)
Engine:
 Version: 25.0.3
 API version: 1.44 (minimum version 1.24)
 Go version: go1.21.6
 Git commit: f417435
 Built: Tue Feb 6 21:14:25 2024
 OS/Arch: linux/amd64
 Experimental: false
containerd:
 Version: 1.6.28
 GitCommit: ae07eda36dd25f8a1b98dfbf587313b99c0190bb
runc:
 Version: 1.1.12
 GitCommit: v1.1.12-0-g51d5e94
docker-init:
 Version: 0.19.0
 GitCommit: de40ad0
PS C:\Windows\System32>

```

Podr amos solicitar tambi n mediante la l nea de comandos, que nos muestre los **comandos disponibles** de Docker, junto con sus **opciones de uso**, mediante **“docker --help”** o **“docker command --help”**.

Adicionalmente, dispondremos de una **aplicaci n tipo escritorio, Dashboard**, desde la que podremos gestionar objetos (contenedores, im genes, ...), a m s alto nivel, a nivel gr fico.

T�tulo	3. Docker. Arquitectura, instalaci�n, administraci�n
Destinatario	1� DAW - Semipresencial
Autor	Pascual Mart�nez
Correo	pmartinez@florida-uni.es



Y dispondremos tambi n de un **icono** desde la **bandeja de sistema**, donde pulsando con el **bot n derecho del rat n**, accederemos a un men  de **accesos r pido**s.

