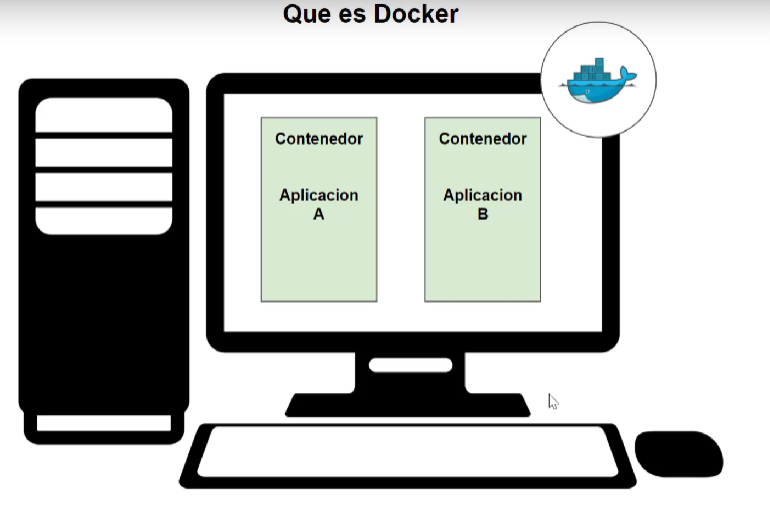
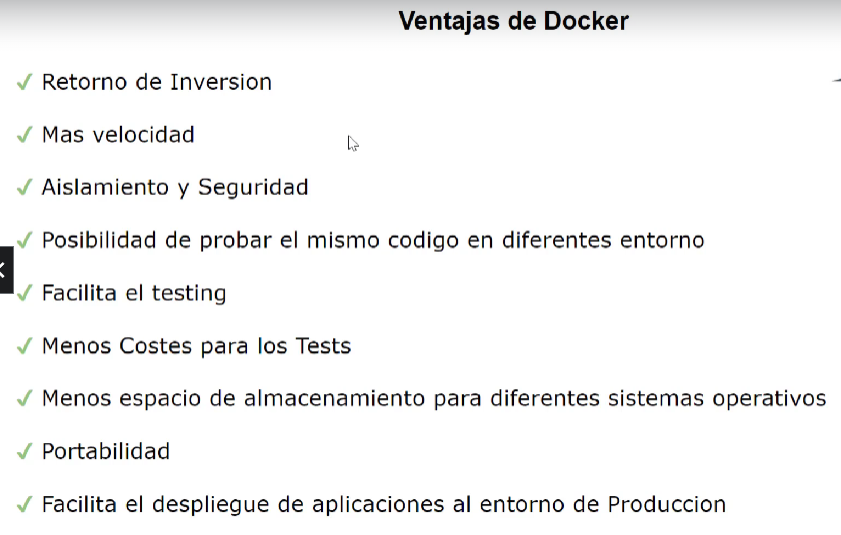
2. Docker Containers

Instalacion de Docker



Sofware de código abierto para enpaquetar app y fácil desplegar en cualquier entorno, utiliza solo recurso de harward



Como funciona Docker

Instalando Docker <https://docs.docker.com/desktop/install/windows-install/> , procedo a descargar el ejecutable

Requisitos para instalar

* Antes Enable the WSL 2 feature on Windows. For detailed instructions, refer to the [Microsoft documentation](https://docs.microsoft.com/en-us/windows/wsl/install-win10).
  + 64-bit processor with [Second Level Address Translation (SLAT)](https://en.wikipedia.org/wiki/Second_Level_Address_Translation)
  + 4GB system RAM
  + BIOS-level hardware virtualization support must be enabled in the BIOS settings. For more information, see [Virtualization](https://docs.docker.com/desktop/troubleshoot/topics/" \l "virtualization).
* Download and install the [Linux kernel update package](https://docs.microsoft.com/windows/wsl/wsl2-kernel).

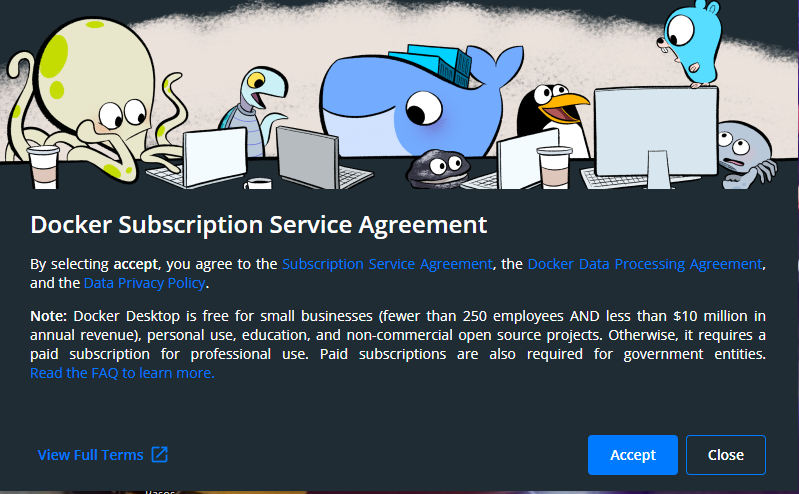
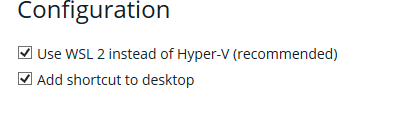
Como salió un texto largo procedo hacer un con el comado wsl –install

Procedo a ejecutar wsl.exe --install -d Ubuntu

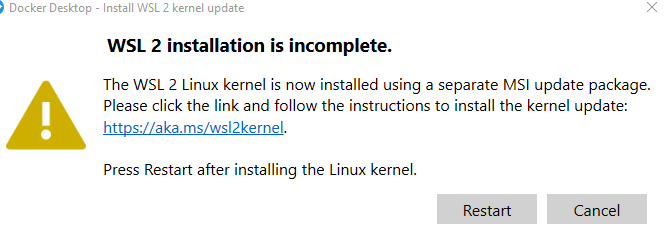
Luego en la ventana de ubunto que se abre automática procedo a agregar un user y un password para el sheel de ubunto ***user user***

Listo WSL se encuentra OK

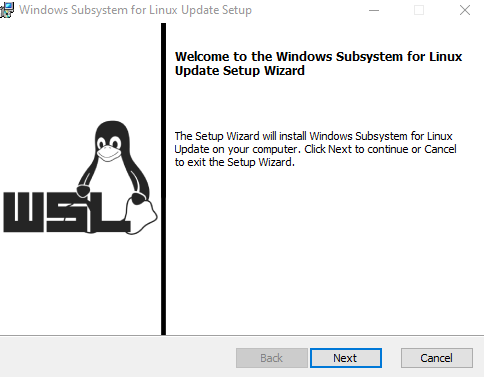
Instalar el ejecutable, y procedo a reiniciar



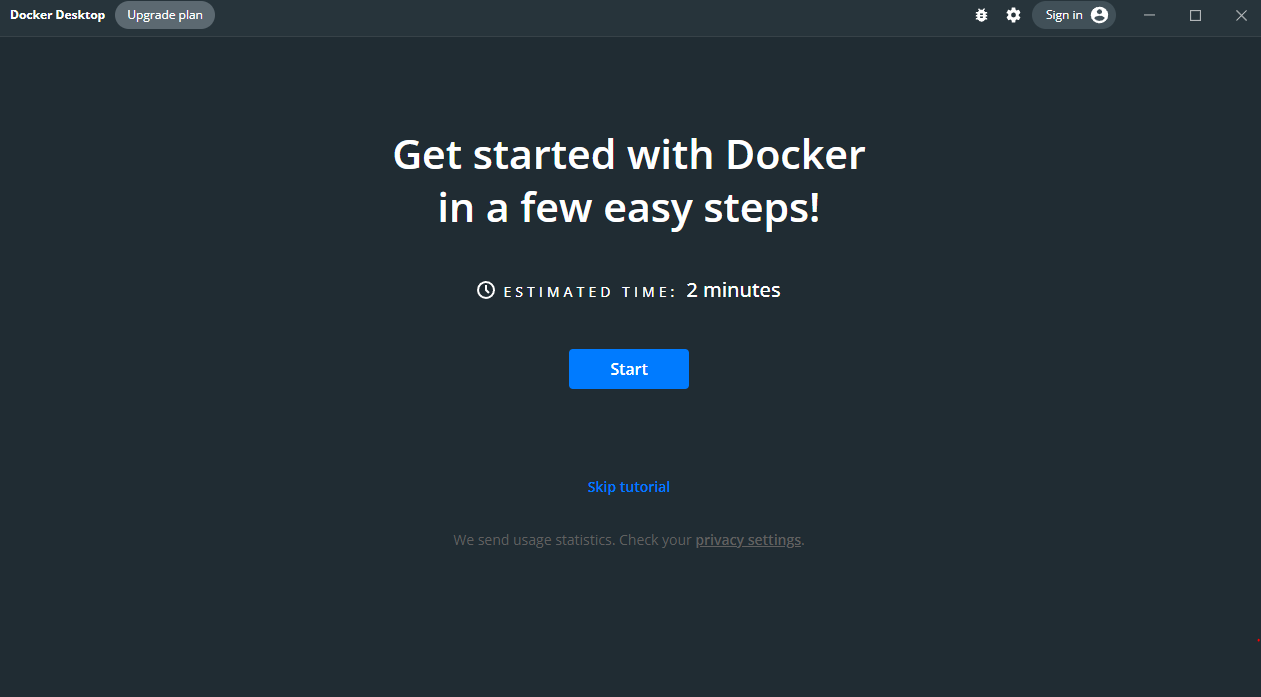
Acepto los términos y listo

Si sale el error

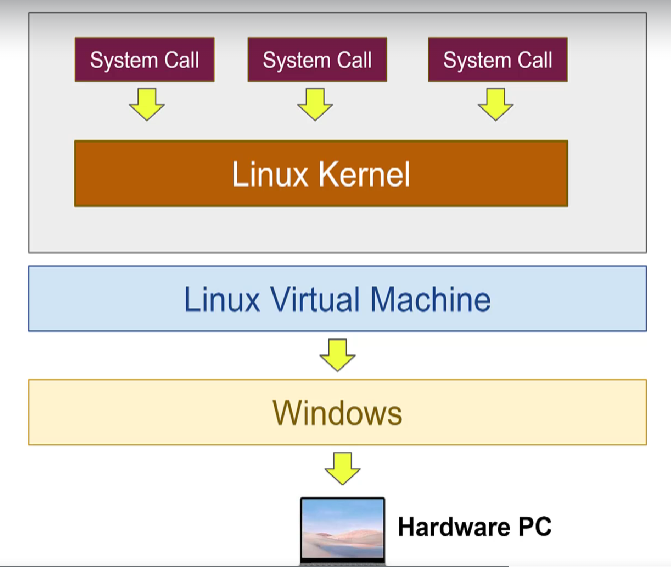
<https://learn.microsoft.com/en-us/windows/wsl/install-manual#step-4---download-the-linux-kernel-update-package> Procedo a desdecargar e instalar el ejecutable de este enlace y listo



Reiniciamos y listo todo OK

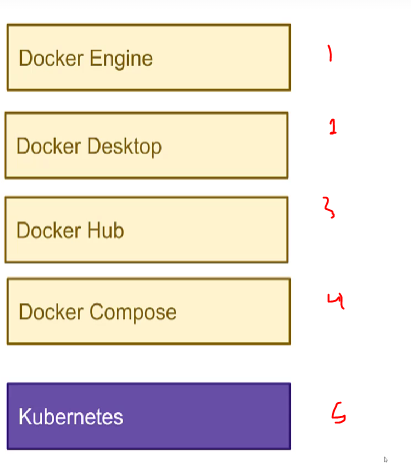


Como funciona docker?



Como se instalo doker en nuestro pc, si es sitema de Linux?

1. Se instala el ambiente para ejecutar el Ubuntu de Linux
2. Asegura que el proceso 1 un este operativo
3. Subir las imágenes a la nube para compartir con los difretntes app coud azurew aws ..
4. Herrramier para administrar containers
5. Administrador de container pero mas op



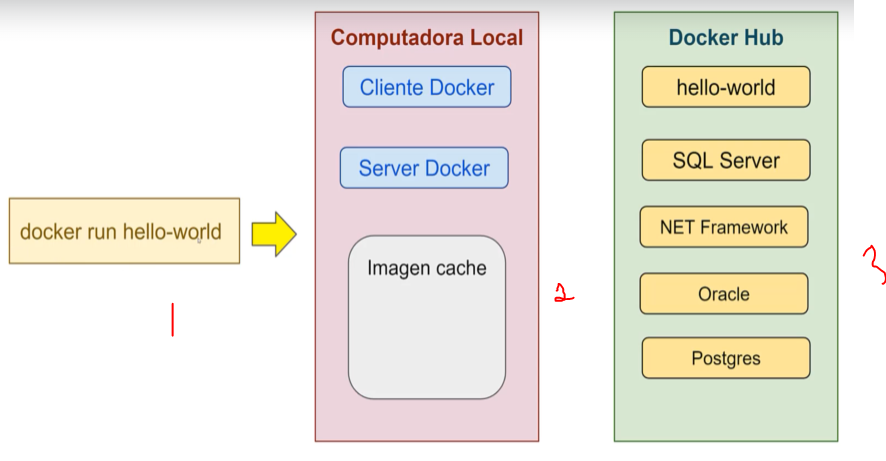
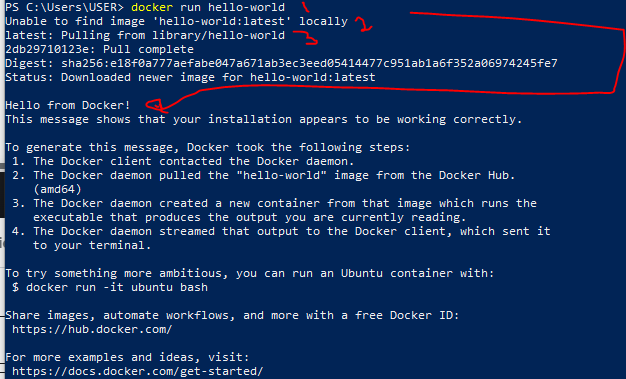
Comandos para ver la versión der Docker

docker -v y docker versión

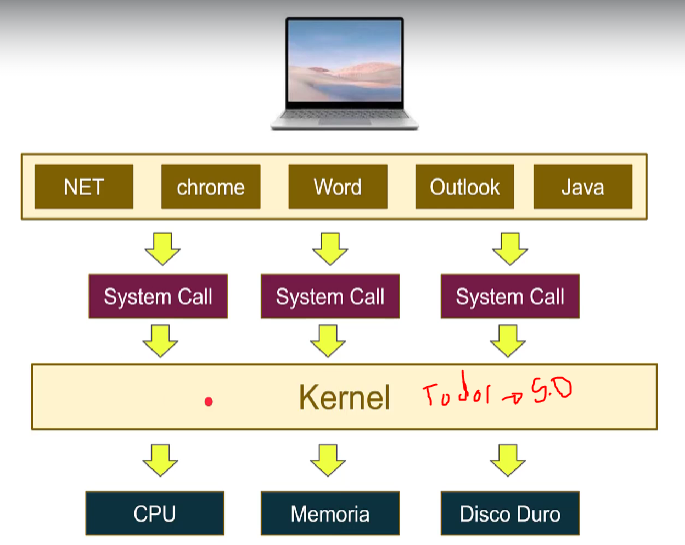
Corremos una hola mundo

Docker run hello-world

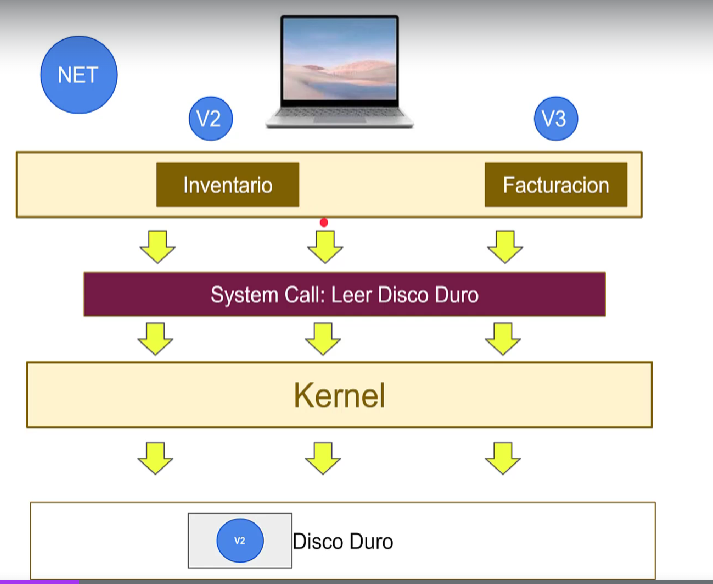
Como no tengo ninguna app con ese nombre lo descarga en automatico de un servidor, para ello veremos el funcionamiento (1. Descarga la imagen 2. Crear el container 3. Cono no tenmos ninguna imagen cache y la busca e el servidor de docker)



Containers en Sistemas Operativos

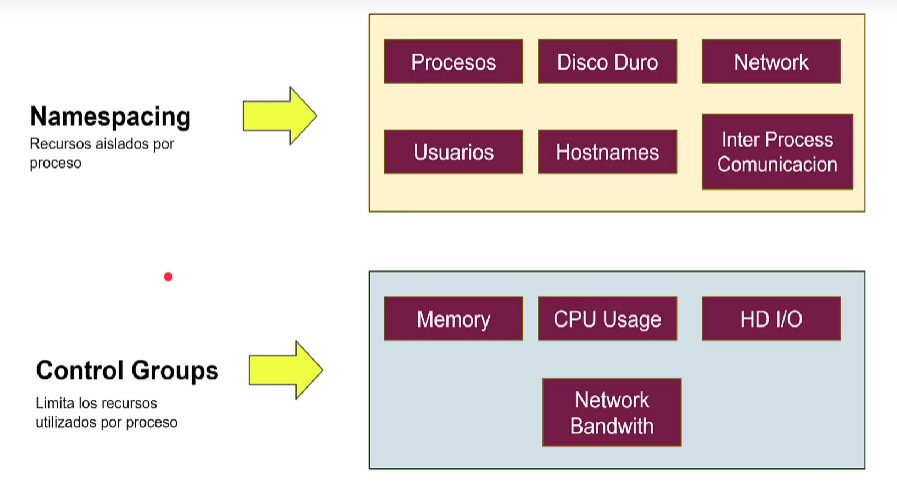


Ejemplo con app NET

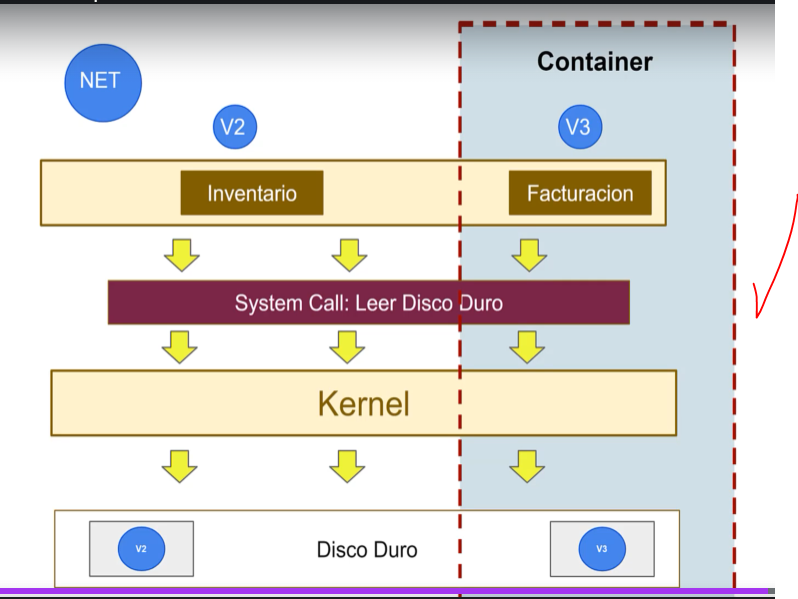


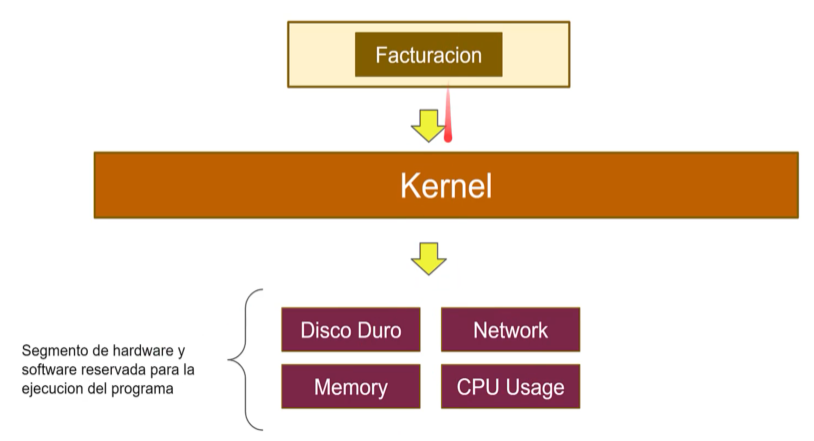
Si en el disco solo esta la V2 esa podrá ser ejecutada (Inventario) mientras que la v3 no

Este problema lo podemos solucionar con una funcionalidad del S.O. llamada name spaicing lo que hace es segmentar o particionar para que se puedan ejecutar la v3 y V2



Lo bueno es que segmentea cualquier sofware o hadware y el control groups es el segmenta los recursos para que la app se ejecute.

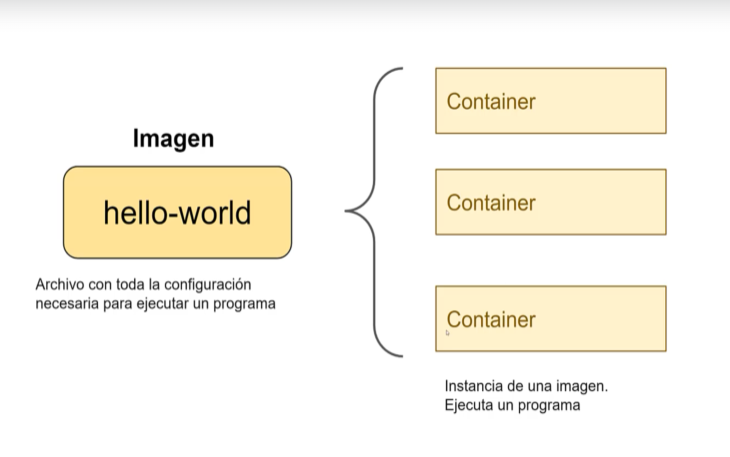


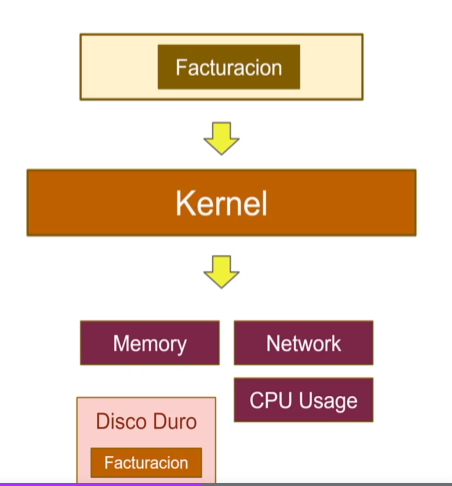
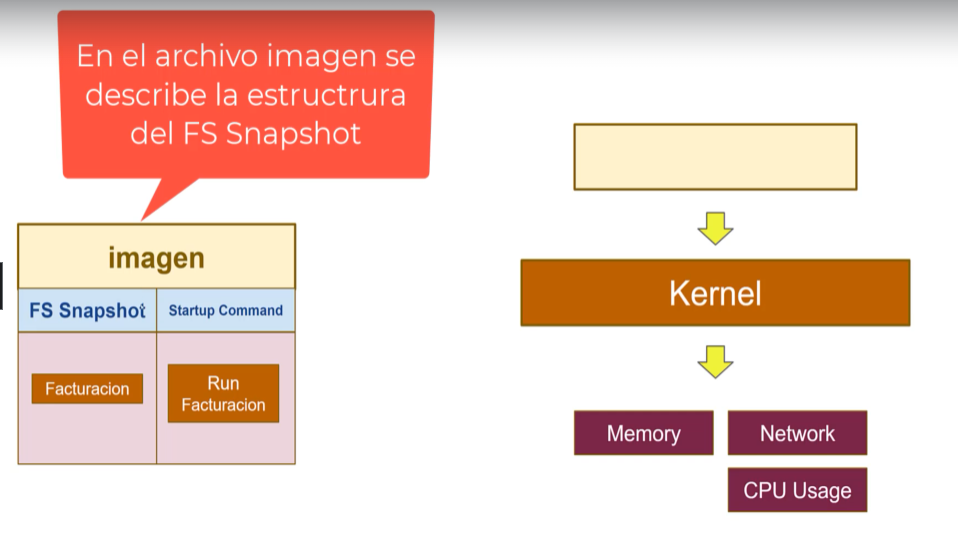
Podemos concluir que un container es un proceso que tiene un grupo de recursos de sofware o hardare para solo para que ejecute la app V3. 

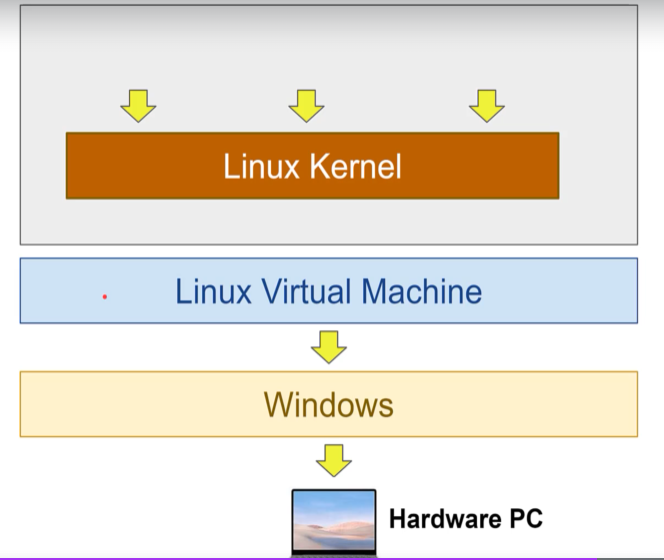
Por ende el kernel me permite tener cantidad de container del mismo kernel del S.O (Sistema Operativo)

Como se crea un docker container?

Un scontainer por una imagen







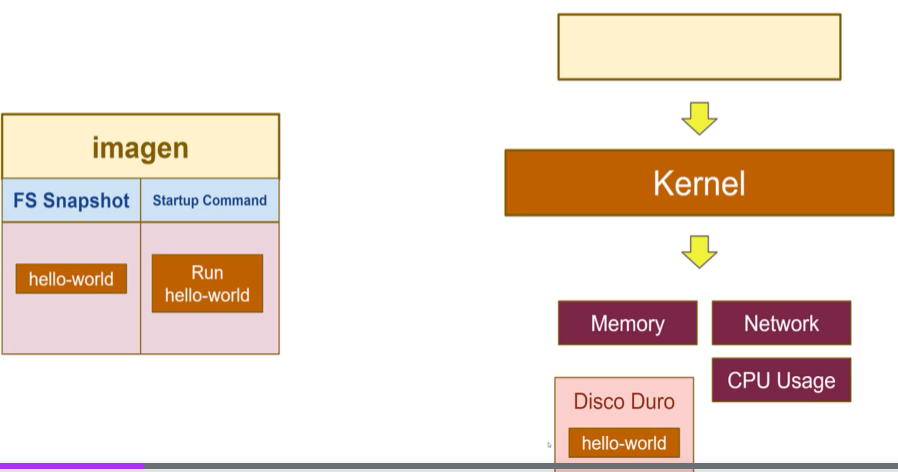
Todos los container que creemos se creara en el virtual machine de Linux

 Comandos en Docker Run



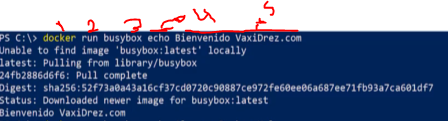
1. Cliente
2. Comando a ejecutar
3. Nombre de la imagen

La tarea que realiza la línea de comando es



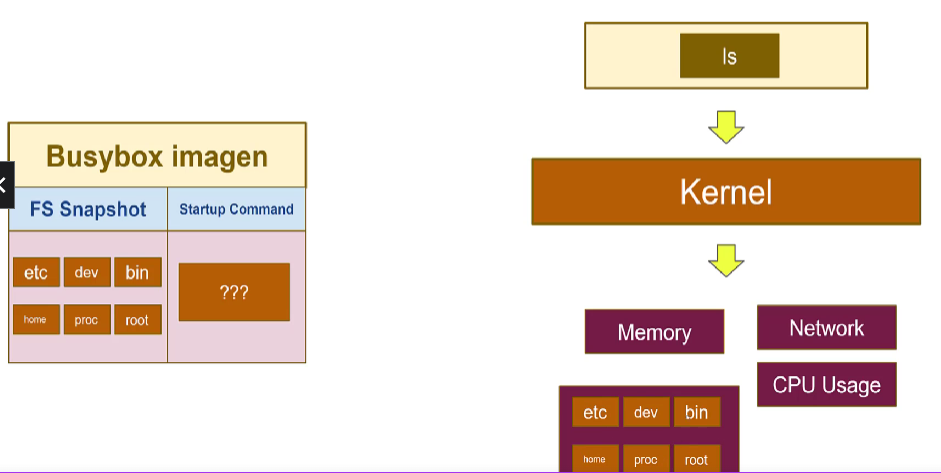
Si quiero crear una nueva imagen imagen con un texto adicional seria así

1. Comando para ejecutar debtro del container
2. String



Comando para conocer la estructura interna de un container

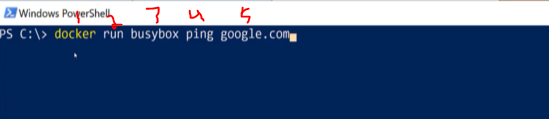




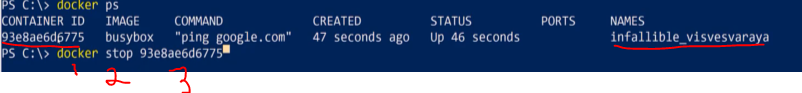
 Comandos para Listar Containers

Como todo container se detiene en automático, entonces para que se siga ejecutando debemos abrir otra terminal y enviar un ping para que se establezca la conexión en tiempo real, comando ps (power shell) me permite ver las características del container que se esta ejecutando

Petición la tente del container

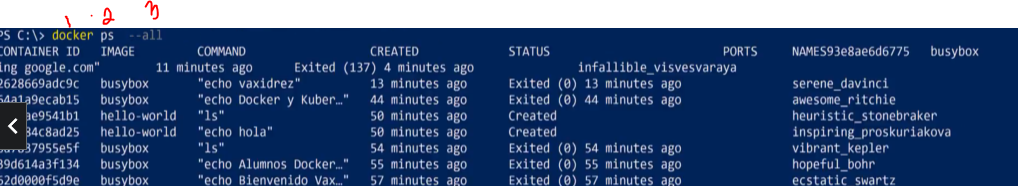


Para poder detener la petición lo podemos hacer sobre la venta que ejecuta la petición CTRL + C o con la venta que esta ejecutando el power Shell (Puede ser con el nombre o el id)

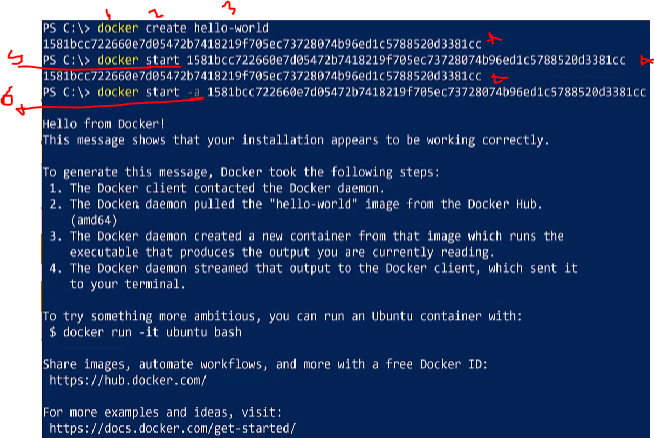


Y fijamos que nos devuelve el nombre el id y en la otra venta se detuvo la petición.

Para ver todos los container EJECUTADOS Y CREADOS en nuestro pc



Ciclo de vida de un container

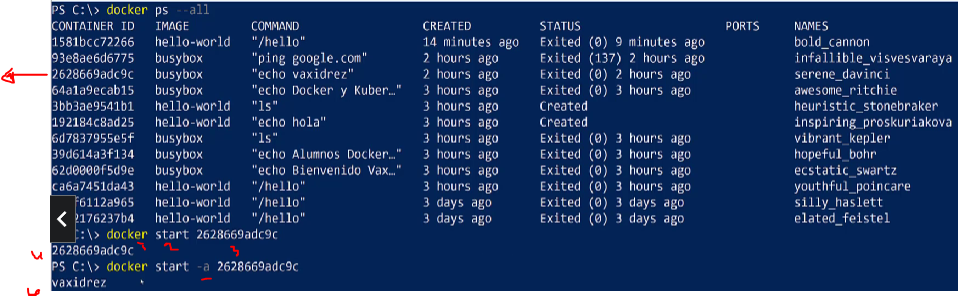


Crear y ejecutar un continer

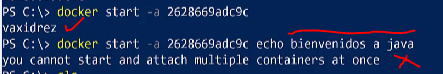
1. Cliente
2. Comando
3. Nombre del container
4. Me devuelve el ID del container creado
5. Comnado
6. ID del Docker que iniciar la ejecución del container y devuelve nuevamente el ID del container
7. Comando del container para ejecutar el container por su ID
8. Salida o string del container

Como vemos el comando RUN ejecutar todos los procesos CREATE – START – LOG

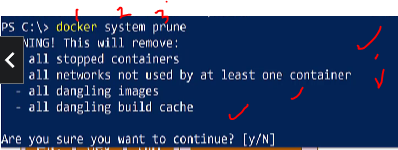
Recordar que en listado para ver todos los container creados, los puedo volver a ejecutar haciendo el llamado por su NAME o ID



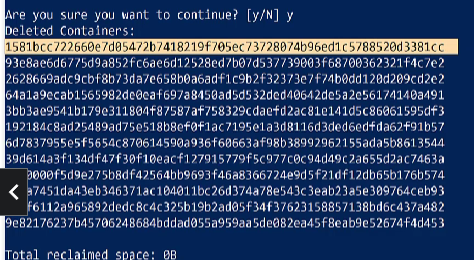
RECORDAR QUE un container ya esta parametrizado, con esto voy que no puedo cambiar la salida o el output para este caso cambiar el vaxidrez por otro texto.



Como eliminar todos los container que esta detenidos DANLING significa IMÁGENEs, muestra todo lo que los tipos de archivos que va a eliminar y con y confirmamos la eliminación.

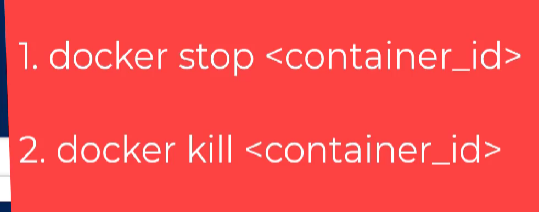


Luego me mostrara todos los container eliminados

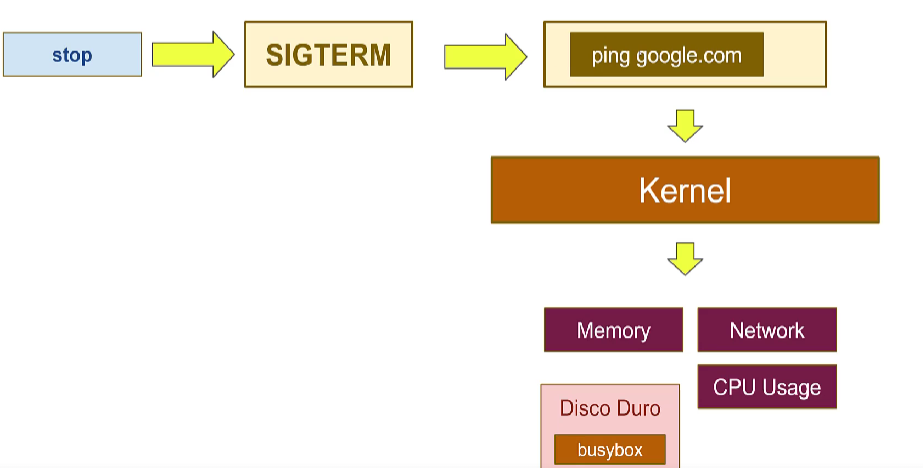


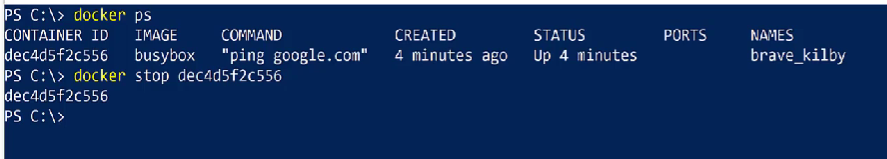
 Stop y Kill Containers

Detener o pagar un procesod e container, existen dos modos

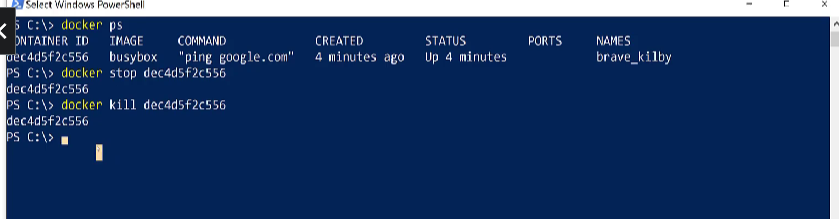


El comando stop limpian y apaga el container. (Es parecido al doc Word con autoguardado forzozo) y luego de 10sgundos como no deteiene el proceso manda el comando KILL claro es para una petición activa





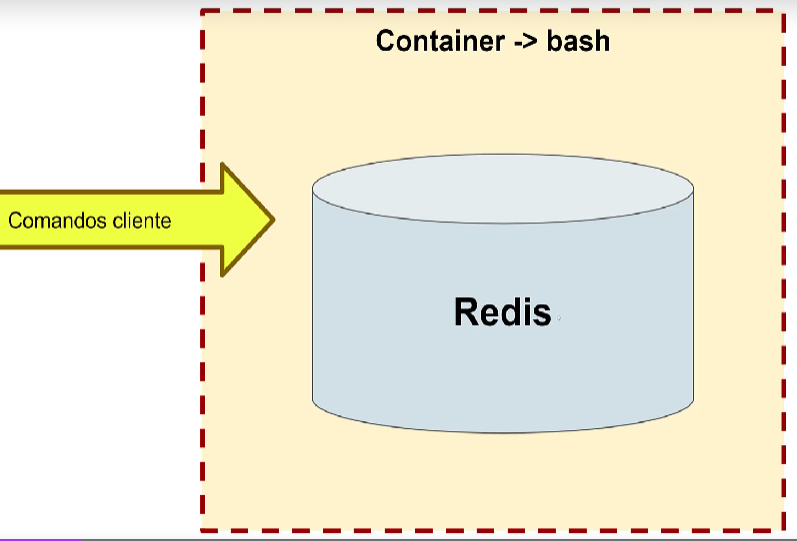
El comando kill mata al instante la conexión



Laboratorio Redis y acceso a Shell

Procesos para producción la app casi siempre estará en un container

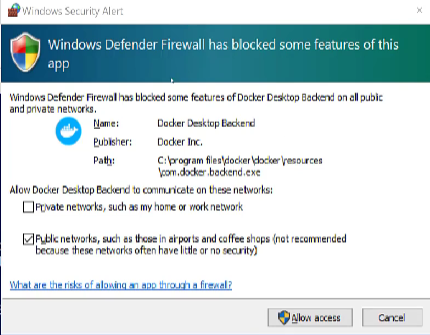
EL lab ejecutara un motor de DB dntro ede un container la tarea es entrar al bash del cotainer para poder hacer peticiones a la DB



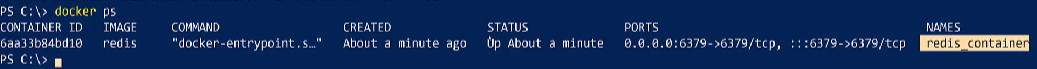
Con el pull bajamos la imagen del servidor relacionado con REDIS 🡪 Docker pull redis

Docker run -d -p 6379:6379 --name redis\_container redis 🡪 la d permitirá ver los logs del contianer y la p es el puerto(Yo le asigno) externo:intereno se asigna un nombre del cliente(Yo le asigno), seguido del nombre de la imagen.

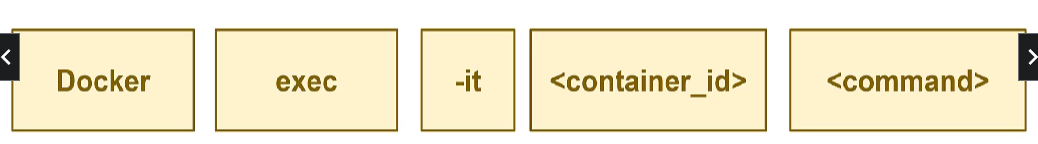
Luego se aceptan los permisos que solicita Windows

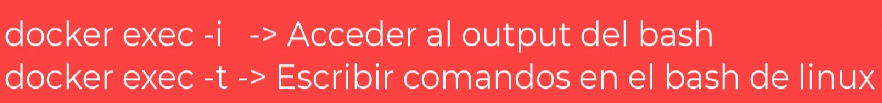


Luego con docker ps validamos las caracteristicas del container



Con esta serie de comandos podremos acceder al bash del container para poder manipular la DB

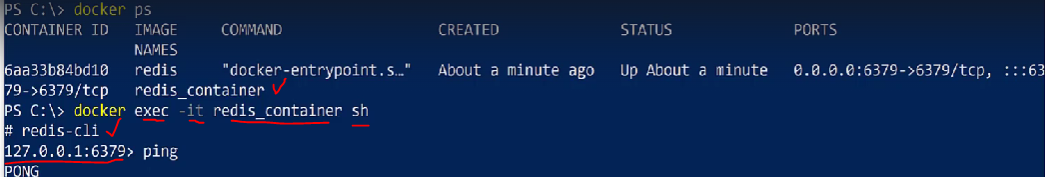




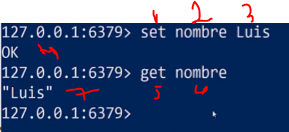
Rl comndo que me permite ingresar al bash es sh (shell)

Importanten escribir el nombre de la imagen-seguido de cli (Interfaz de linea de comandos) y nos arroja la ruta del servidor local con el púrto, de este modo ya podemos operar la DB

Escribirmos un ping en lservidor para verrficar que todo este perfecto y nos devolvera un pong

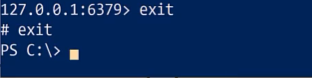


En este caso las consutlas a la DB se realizan por medio de KEYS (Llaves primarias )



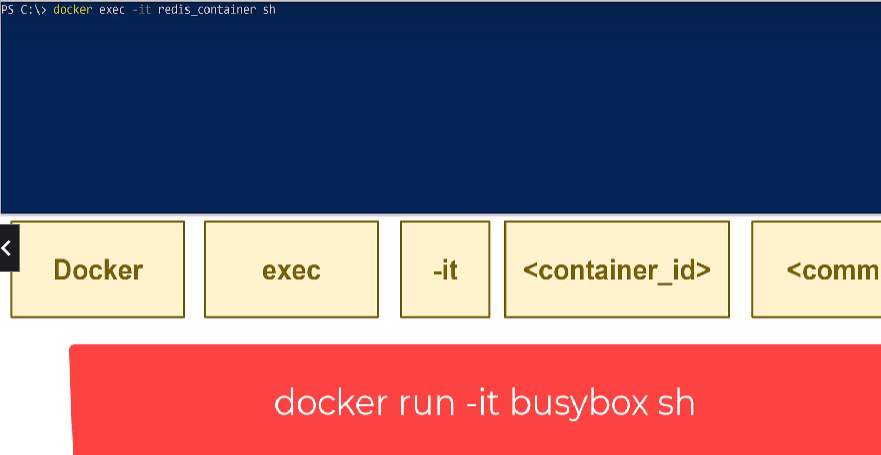
1. Creamos la llave NOMBRE con la data LUIS
2. Llave
3. Data
4. Resuesta que la key a sido creada
5. Llamar la key
6. Nombre de la key
7. Data de la Key

Procedo a salir del servidor con exit y luego el numera me indica que esto en el ps de Linux y le escribo exit para llegar al ps de Windows

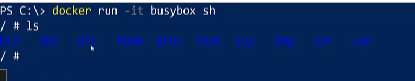


 Acceso a shell con Docker Run

Aqui estamos abriendo el Shell del container



No es recomenda utiiozar este comando en container que ya contengan información, puede psar que el Shell sh remplace información de la DB u otros servicios del mismo. En este aso el busybox esta vacio como vemos el # es el acceso al Shell del container



Creacion de imagenes con Docker

Por ahora todas las imágenes ejecutadas se han descargado del servidor para este caso crearemos una imagen. Proceso

