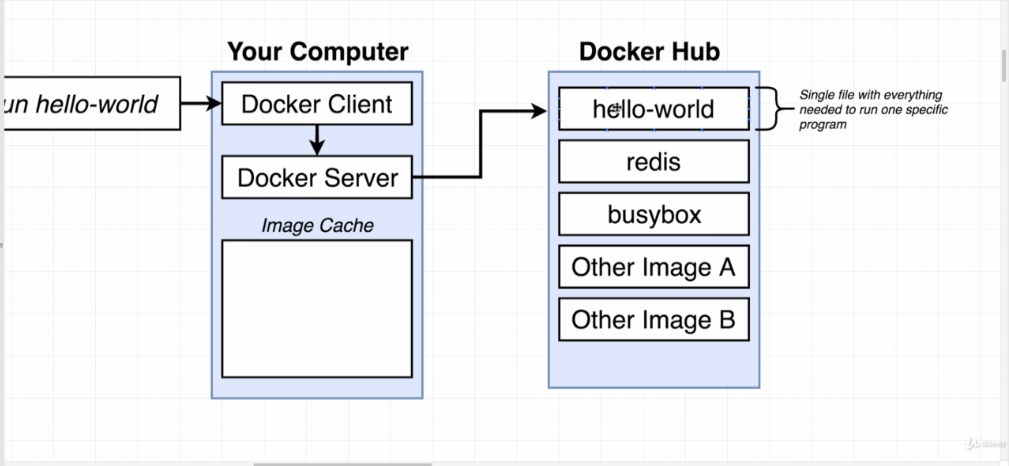
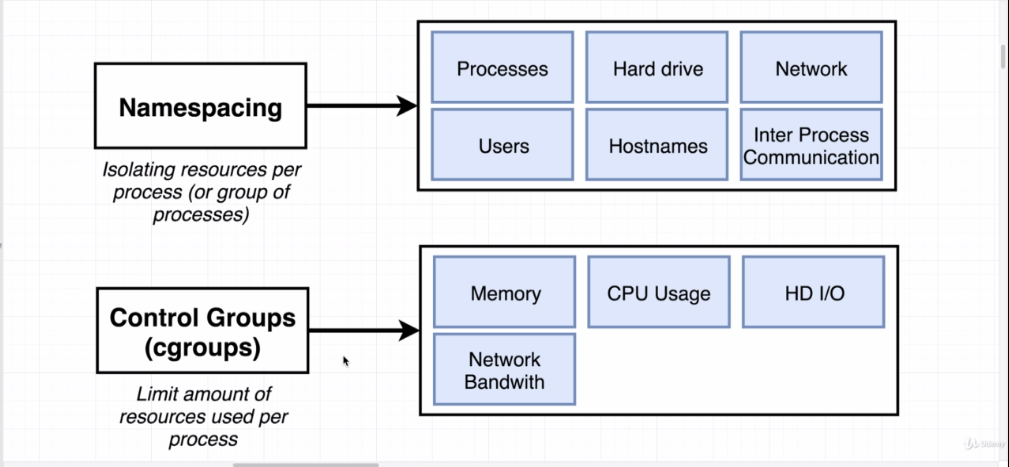
* Docker image is a software which is either locally stored or on the Docker hub
* A container is an instance of the docker image.
* An image can be run on multiple containers
* *Docker run “image-Name” –* this command tries to check if the image exists locally

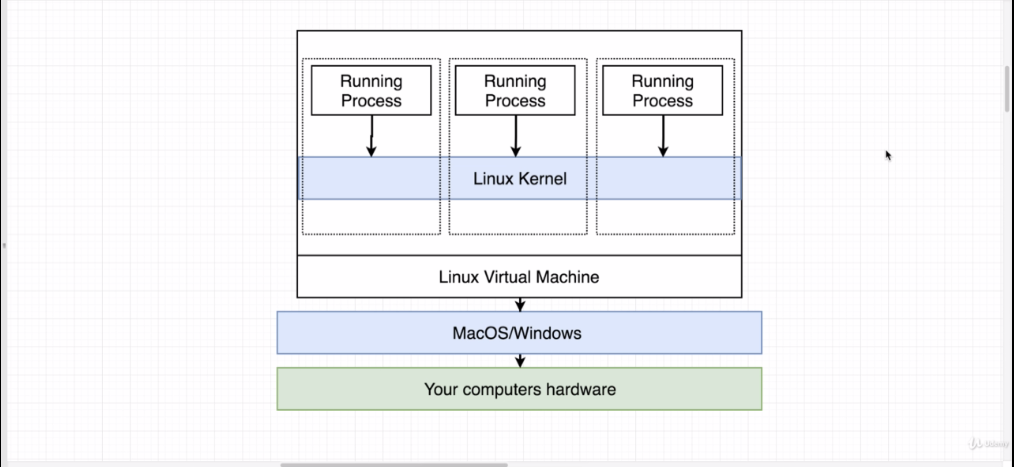


The docker CLI tries to find the image in the local cache, if not found, downloads it from docker hub and stores a copy in the local image cache.

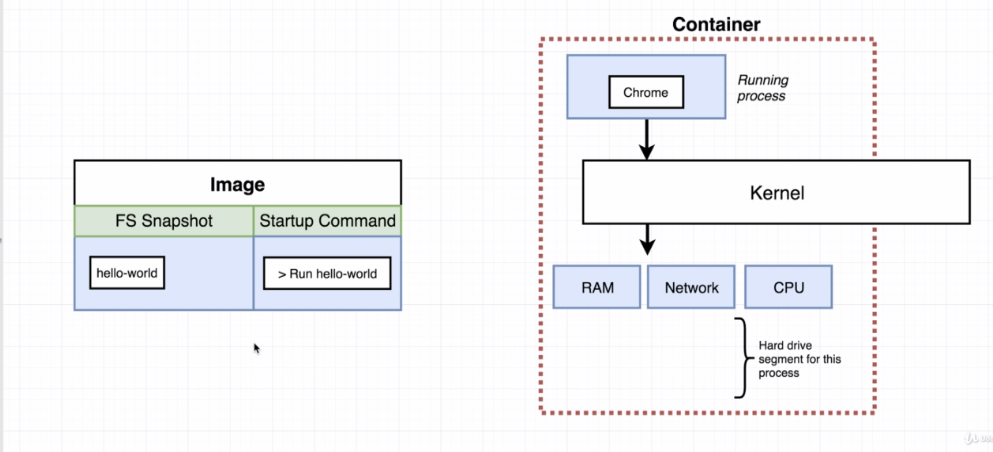
Namespacing & Control groups – used to group the resources which can be used by a process. These belong to Linux OS but not windows and mac

-- *A container is a package of the application along with kernel which is allocated with hardware resources*

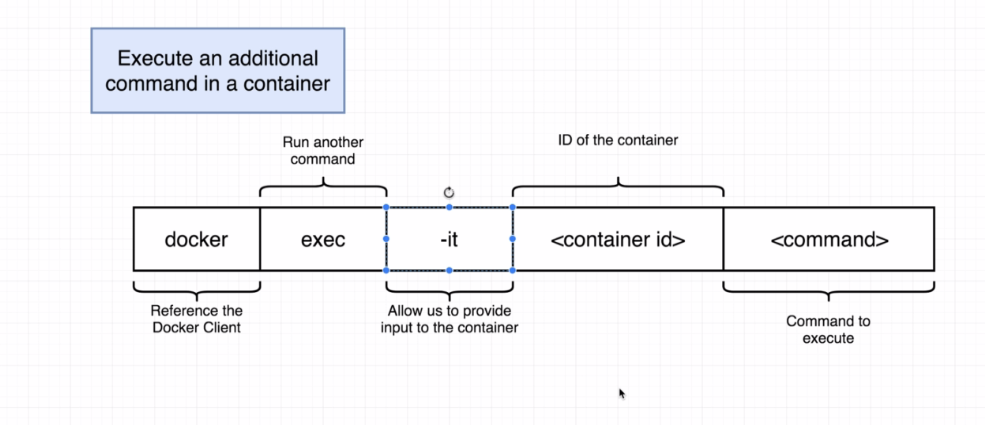
A layer of the Linux virtual machine is installed upon the Windows Operating system for the containers to be installed and run. As docker is run on the Linux server and will continue to work until the docker is running.



Docker image with the container:



* Docker run <Image-Name> <Command>
  + <Command> should be part of the image file system. In layman's terms, the function being called must exist as part of the image.
* docker ps < - -all >
  + to know the containers which are running currently on the local machine.
  + –all is optional and shows all the containers which are run on the machine.
* Docker run is a combination of docker create and docker start.
  + Docker start -a <container ID> --- -a is used to capture the output from the container
* docker system prune
  + Deletes the containers which are present on the hard disc waiting to be started
* docker logs <container ID> -- used to check the output produced by the docker start.
* In order to stop a running container
  + docker stop/kill <container ID>, both the stop and kill commands whereas stop gives some time for the running process to shut down while the kill command stops it immediately

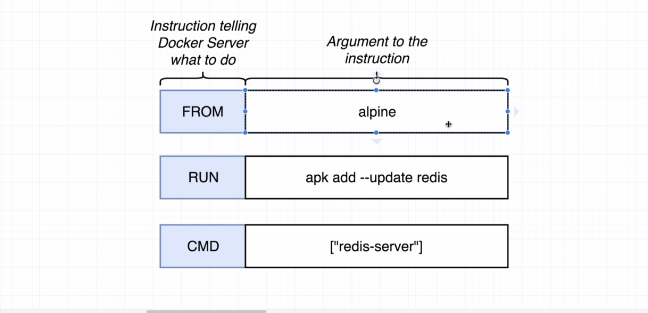


To stop all the containers running on the machine

***docker container stop $(docker container ls -aq)***

🡪 docker build .

* used to build the docker image using the yml file



These commands were used to run Redis server on a container.

* Apk is a keyword used to fetch package. Apk is part of alpine, it is similar to NuGet.
* Docker uses the cached images when we are trying to rebuild the image file
* When we are running multiple images in our custom image, then until the order of images is changed the images will be fetched from cache, but once the sequence is changed, the images will be redownloaded and recreated.

**RUN apk add –update gcc**

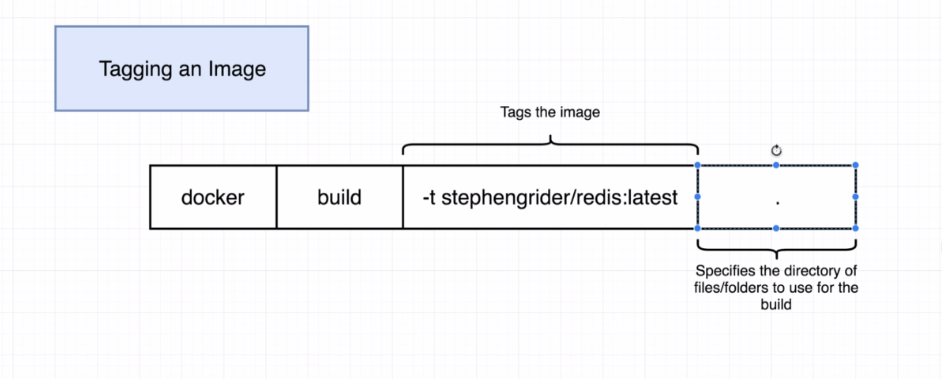
**RUN apk add –update redis**

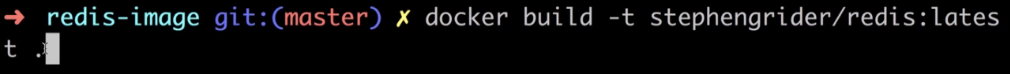
Is different from

**RUN apk add –update redis**

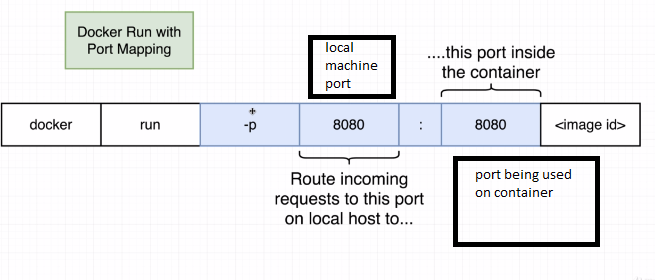
**RUN apk add –update gcc**

**🡪**Tagging the images

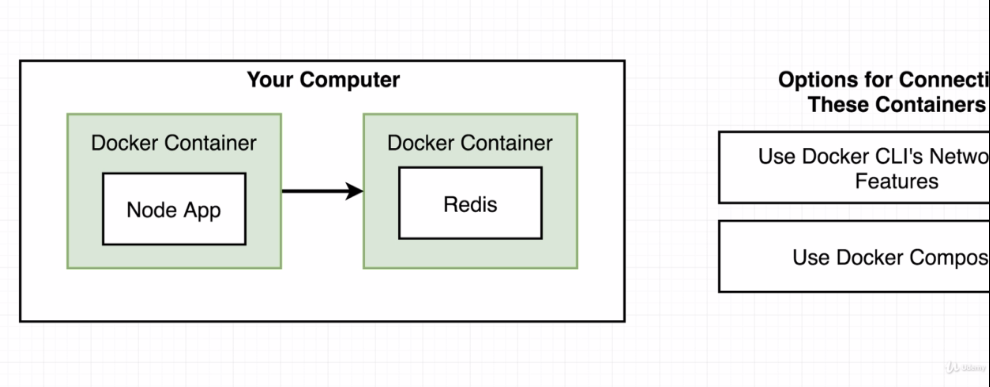




* Port mapping



* Docker compose??
* In the visits project we use the docker compose for the following reason, to create a container with single container for all instances of other container services



* **docker-compose up**
* **docker-compose up –build** (if the image is not built)
* **docker- compose up -d** (if you want to run the docker container in the background)
* **docker-compose down** (to stop all the container run by docker-compose)
* in order to keep the container running whenever it crashes, we have restart policies in the docker-compose file with options as below  
  