**Docker Walk 101: Building Containers**

*Posted On August 22, 2019*

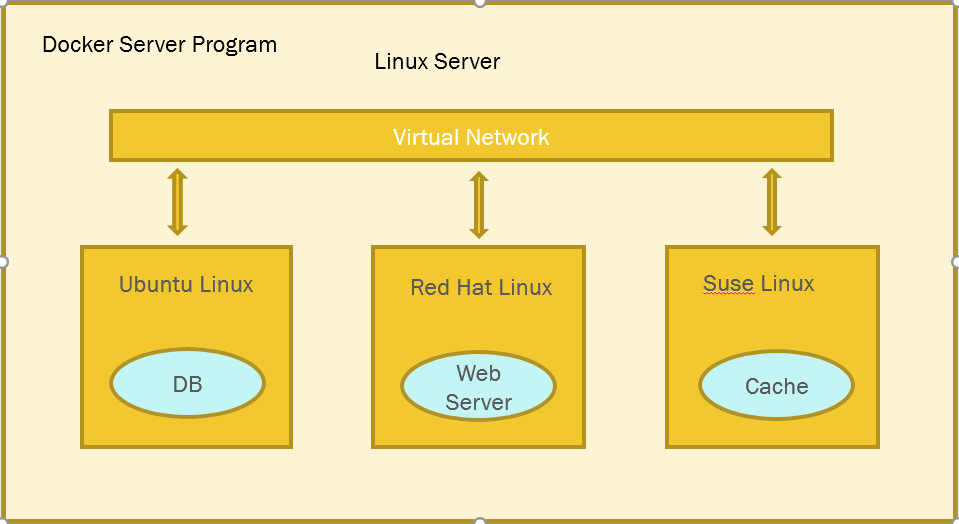
Docker carves up a system/computer into a sealed container that run code.

Containers are designed to be portable so that it can be shipped from one place to another and docker gets these work of getting to and from system. A container is a sealed unit of software.

Docker builds these containers. A container includes:-

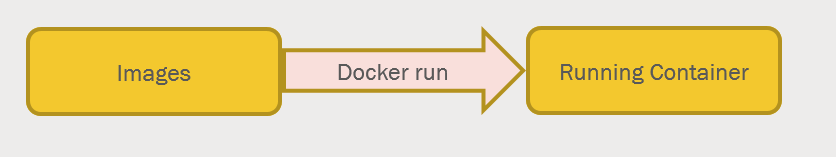
Code  
Configs  
Dependencies  
Networking  
Operating System  
Processes

There can be different containers built on different Linus Systems connnected by Virtual Network. One container can contain Database running on a Ubuntu Linux, connected to a another web server through container running on a Red Hat Linux and that web server might also be talking to a Suse Linux having container to handle cache.



Docker is a program which manages all of these. Set it up, manages these and remove it when it is no longer needed. Docker has a program which builds containers from code. It takes your code along with its dependencies and bundles it up and then seals it into a container. It then also works as service to distribute containers across internet.

**Run processes in containers**

Once you have installed docker, run command :-  
**$ docker run -ti –rm ubuntu bash**  
-ti helps it be to interactive. It refers to Terminal Interactive. Ubuntu environment is setup and shows :  
root@dgdk2j45j: /# — Where charachters after root@ represents container ID that is currently running in.  
**$ docker info** — Provides lots of information about Docker.  
**$ docker run hello-world** –It will pull up latest image from Docker Hub if not available locally.  
  
$ **docker images** — the command to look into docker images.  
  
The docker run command takes an image and turns it to running container with a process doing set of task/s.  
  
  
ti command helps running command by typing on a keyboard inside an image.  
For exiting the command shell, press Conrol D.  
**$ docker ps** — helps to take a look at running images.  
**$ docker run** — starts a container by giving an image name and a process to run in that container. When the process exits, the container work is done.