Introduction

USWAY©
REINVENT YOURSELF
Students, write your response!

Pear Deck Interactive Slide
Do not remove this bar

Table of Contents

- What is Docker?
- What is Container?
- Before Docker
- Docker vs. VMs
- Docker Architecture
- Images and Containers

CLARUSWAY©
WAY TO REINVENT YOURSELF

1

What is docker ???

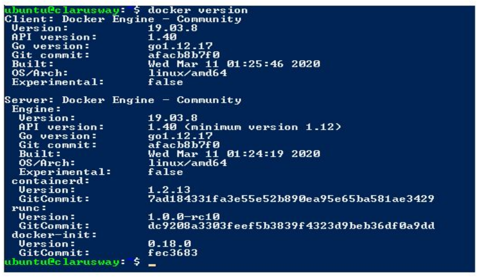
CLARUSWAY©
WAY TO REINVENT YOURSELF

What is Docker?

"DOCKER" refers to several things. This includes an open-source community project which started in 2013; tools from the open-source project; Docker Inc., the company that is the primary supporter of that project; and the tools that the company formally supports.

- Docker as a "Company"
- Docker as a "Product"
- Docker as a "Platform"
- Docker as a "CLI Tool"
- Docker as a "Computer Program"

docker



```
ubuntu@clarusway: $ docker version
Client: Docker Engine - Community
 Version:           19.03.8
 API version:       1.40
 Go version:        go1.12.17
 Git commit:        afacb8b7f0
 Built:             Wed Mar 11 01:25:46 2020
 OS/Arch:           linux/amd64
 Experimental:      false

Server: Docker Engine - Community
 Engine:
  Version:          19.03.8
  API version:      1.40 (minimum version 1.12)
  Go version:       go1.12.17
  Git commit:       afacb8b7f0
  Built:            Wed Mar 11 01:24:19 2020
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.2.13
  GitCommit:        7ad184331fa3e55e52b890ea95e65ba581ae3429
 runc:
  Version:          1.0.0-rc10
  GitCommit:        dc9208a3303feef5b3839f4323d9beb36df0a9dd
 docker-init:
  Version:          0.18.0
  GitCommit:        fec3683
ubuntu@clarusway: $ _
```

CLARUSWAY©
WAY TO REINVENT YOURSELF

What is Docker?

CLARUSWAY©
WAY TO REINVENT YOURSELF

2 What is Container?

CLARUSWAY©
WAY TO REINVENT YOURSELF

What is Container?

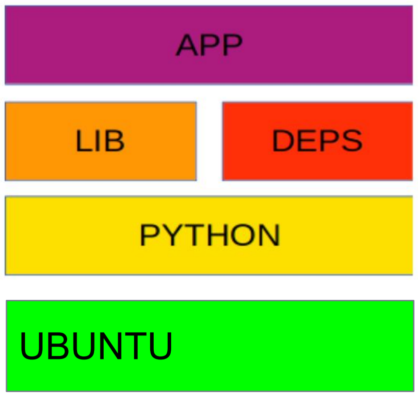
Imagine you're developing an python application. In order to do so you will setup an environment with python installed in it. You do your work on a laptop and your environment has a specific configuration. The application you're developing relies on that configuration and is dependent on specific libraries, dependencies, and files.

Once the application is developed, it needs to be tested by the tester. Now the tester will again set up same environment.

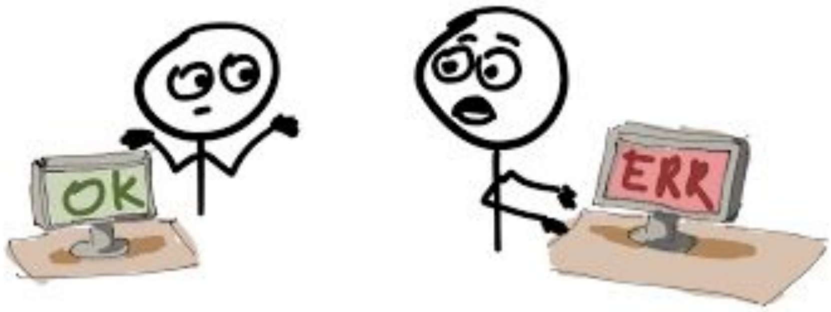
Once the application testing is done, it will be deployed on the production server. Again the production needs an environment with libraries, dependencies, files and python installed on it.

How do you make your app work across these environments, pass quality assurance, and get your app deployed without massive headaches, rewriting, and break-fixing?

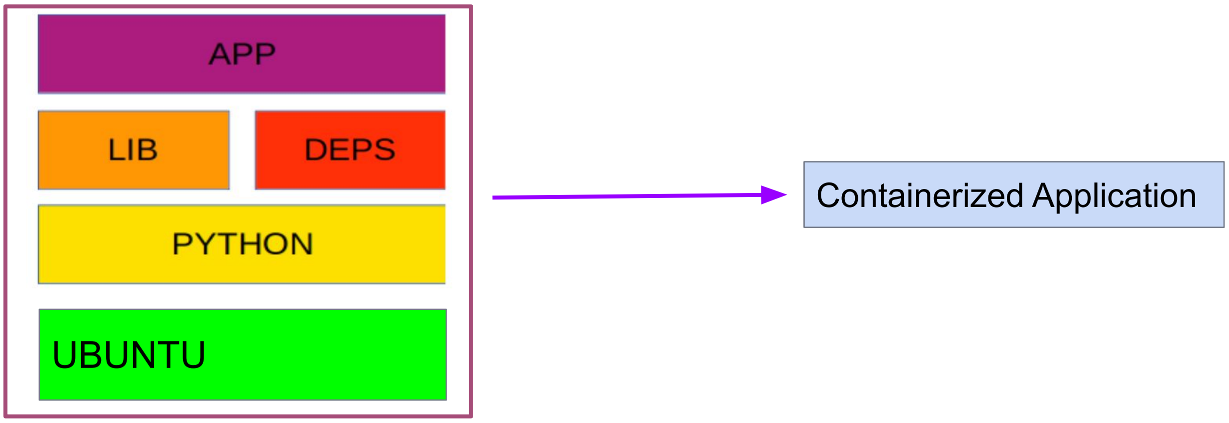
What is Container?



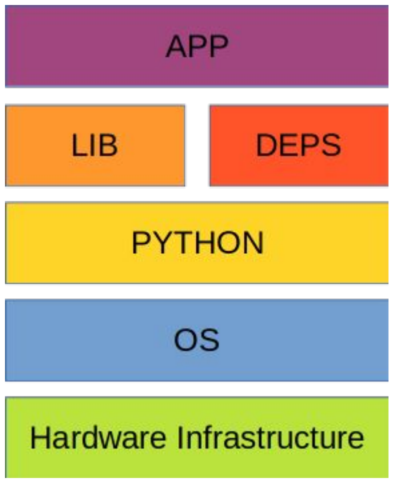
What is Container?



What is Container?

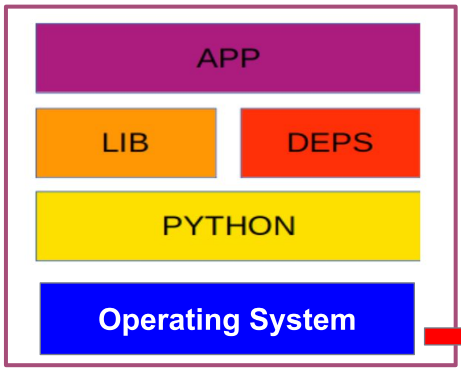


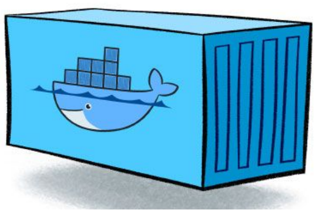
What is Container?



Kernel, Media Player, Browser, Calculator, Solitaire, GUI, Calendar, Paint ...

What is Container?

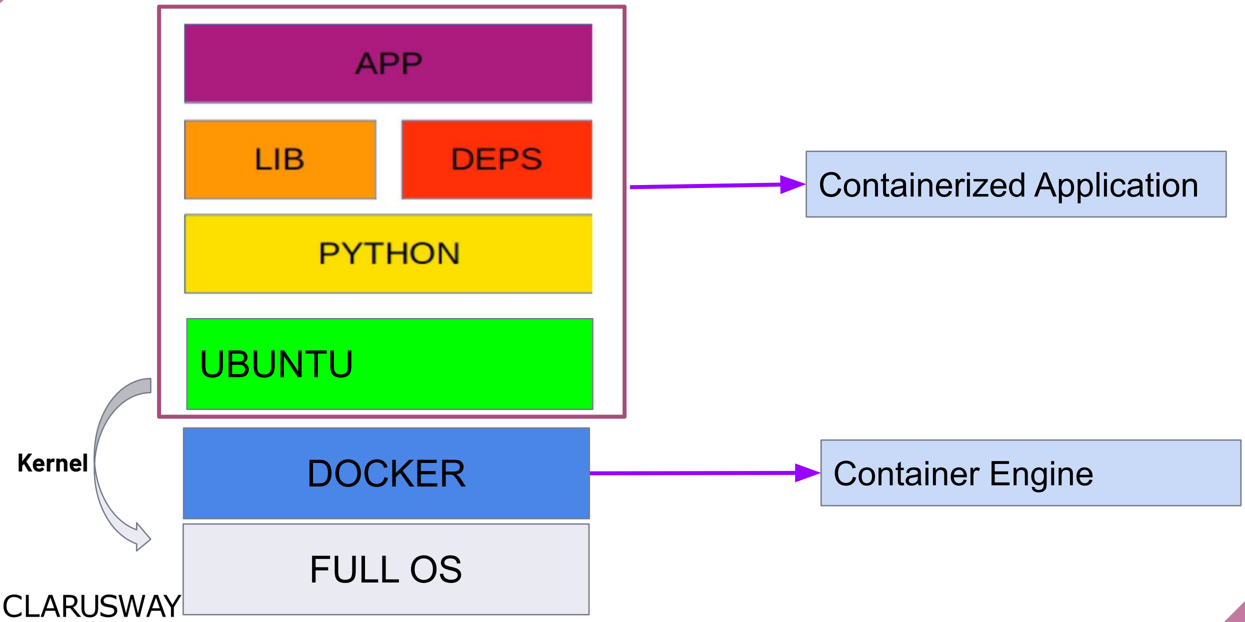




Container

Just required programs for application (without kernel)

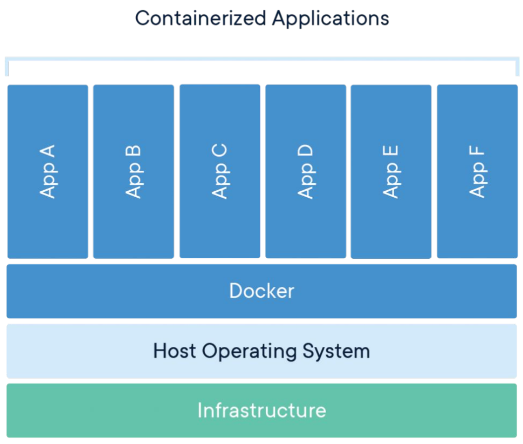
What is Container?



CLARUSWAY
WAY TO REINVENT YOURSELF

What is Container?

A **container** is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another.



CLARUSWAY©
WAY TO REINVENT YOURSELF

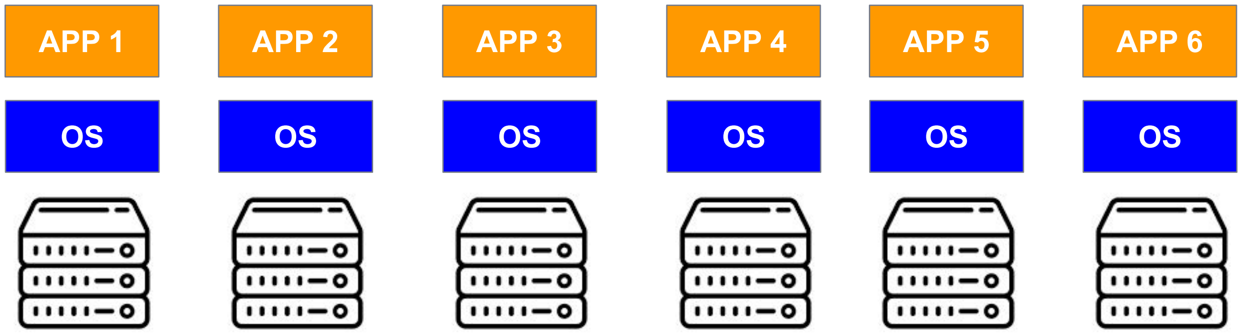
3

Before Docker

Bare Metal

OS	= 1.5 CPU + 1.5 GB RAM
APP	= 0.5 CPU + 0.5 RAM

18 CPU
18 GB RAM

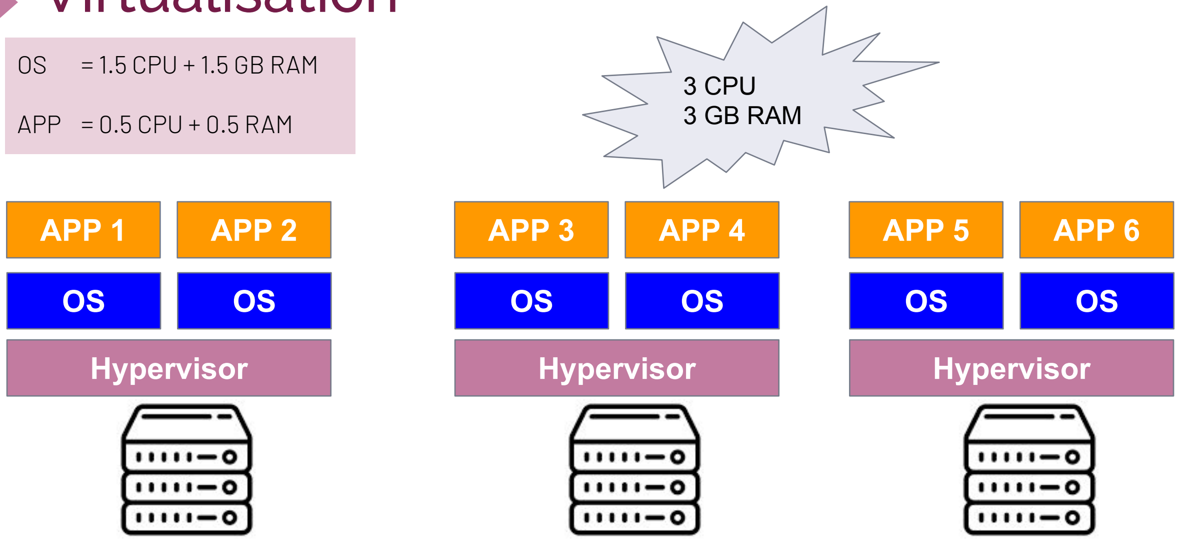


HARDWARE = 5 CPU + 5 GB RAM

Virtualisation

OS = 1.5 CPU + 1.5 GB RAM

APP = 0.5 CPU + 0.5 RAM



HARDWARE = 5 CPU + 5 GB RAM

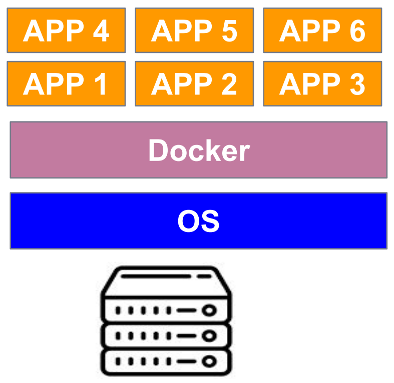
CLARUSWAY©
WAY TO REINVENT YOURSELF

Bare Metal

Container

OS = 1.5 CPU + 1.5 GB RAM

APP = 0.5 CPU + 0.5 RAM



HARDWARE = 5 CPU + 5 GB RAM

CLARUSWAY©
WAY TO REINVENT YOURSELF

What is Container?

CLARUSWAY©
WAY TO REINVENT YOURSELF

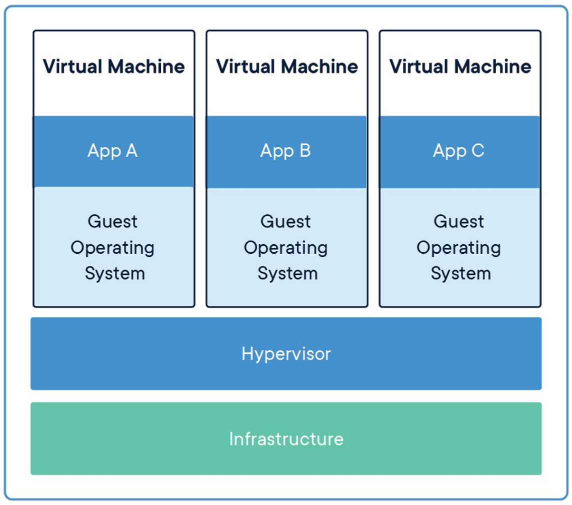
4 Docker vs. VMs

CLARUSWAY©
WAY TO REINVENT YOURSELF

Docker vs. VMs

A virtual machine (VM) is software that runs programs or applications without being tied to a physical machine.

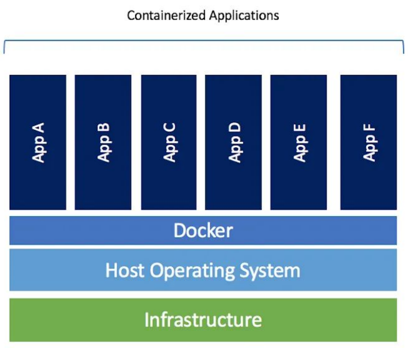
Virtual Machines are built over the physical hardware, there is a hypervisor layer which sits between physical hardware and operating systems.

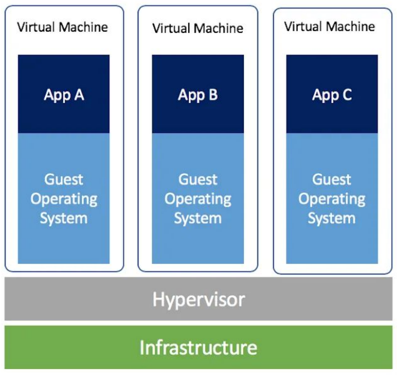


CLARUSWAY©
WAY TO REINVENT YOURSELF

Docker vs. VMs

Unlike virtual machines where hypervisor divides physical hardware into parts, Containers are like normal operating system processes.





CLARUSWAY©
WAY TO REINVENT YOURSELF

Docker vs. VMs

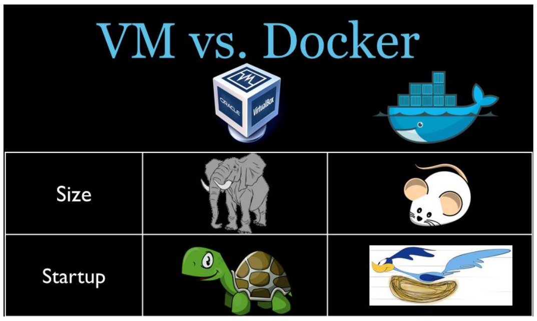
Virtual Machine

Containers

Docker containers are executed with the Docker engine rather than the hypervisor. Containers are therefore smaller than Virtual Machines and enable faster startup with better performance, less isolation and greater compatibility possible due to sharing of the host's kernel. Hence, it looks very similar to the residential flats system where we share resources of the building.

Docker vs. VMs

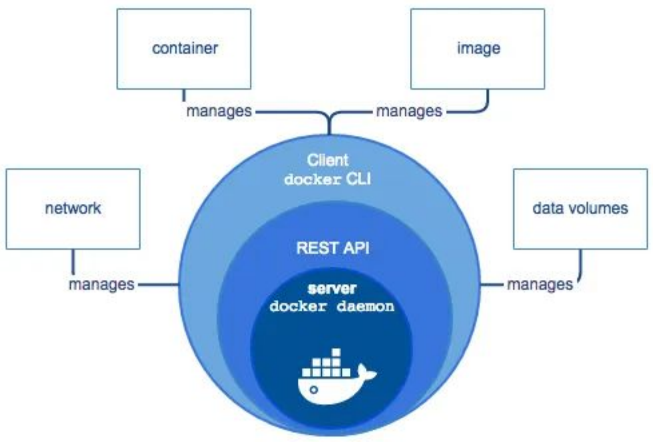
Docker	Virtual Machines
All containers share the same kernel of the host	Each VM runs its own OS
Containers instantiate in seconds	Boots uptime is in minutes
Containers are lightweight (KBs/MBs)	VMs are of few GBs
Less resource usage	More resource usage
Can run many Docker containers on a laptop.	Cannot run more than a couple of VMS on an average laptop



5 Docker Architecture

Docker Architecture

Docker uses a client-server architecture. The Docker client talks to the Docker daemon, which does the heavy lifting of building, running, and distributing your Docker containers. The Docker client and daemon can run on the same system, or you can connect a Docker client to a remote Docker daemon. The Docker client and daemon communicate using a REST API, over UNIX sockets or a network interface.



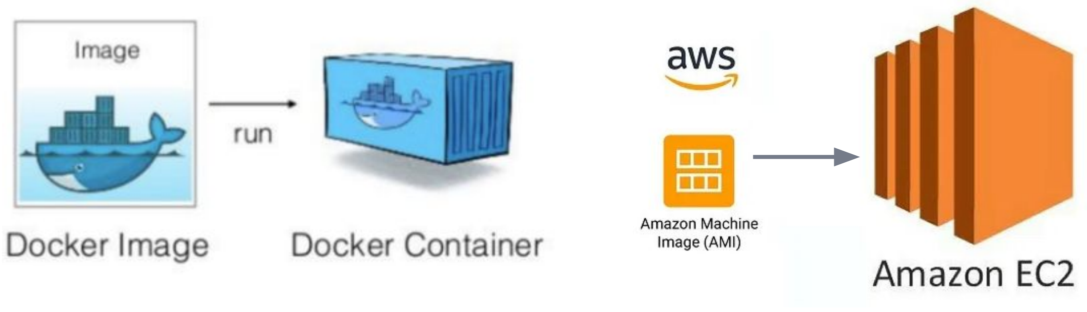
6

Images and Containers

CLARUSWAY©
WAY TO REINVENT YOURSELF

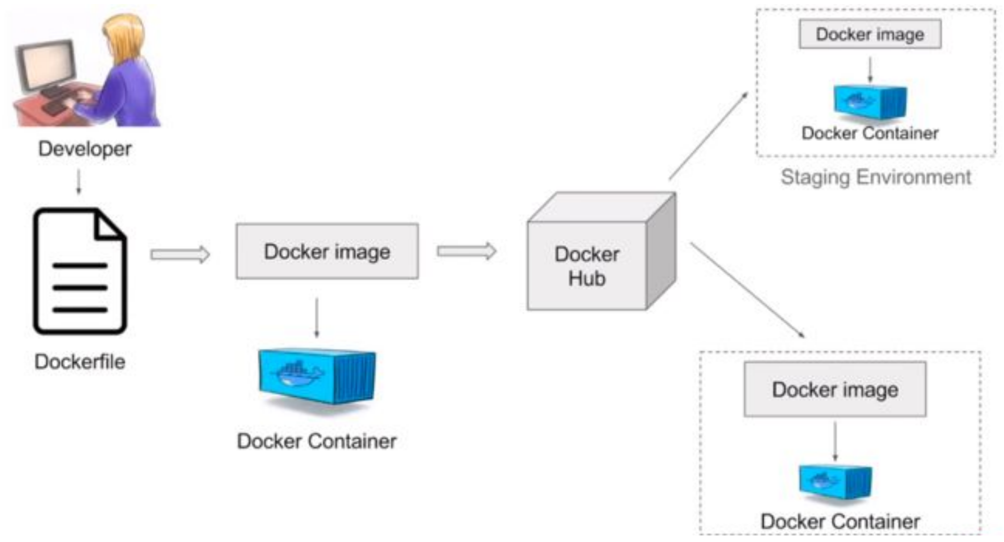
Images and Containers

- An image is a read-only template with instructions for creating a Docker container.
- A container is a runnable instance of an image.



CLARUSWAY©
WAY TO REINVENT YOURSELF

Images and Containers



CLARUSWAY©
WAY TO REINVENT YOURSELF

THANKS!

Any questions?

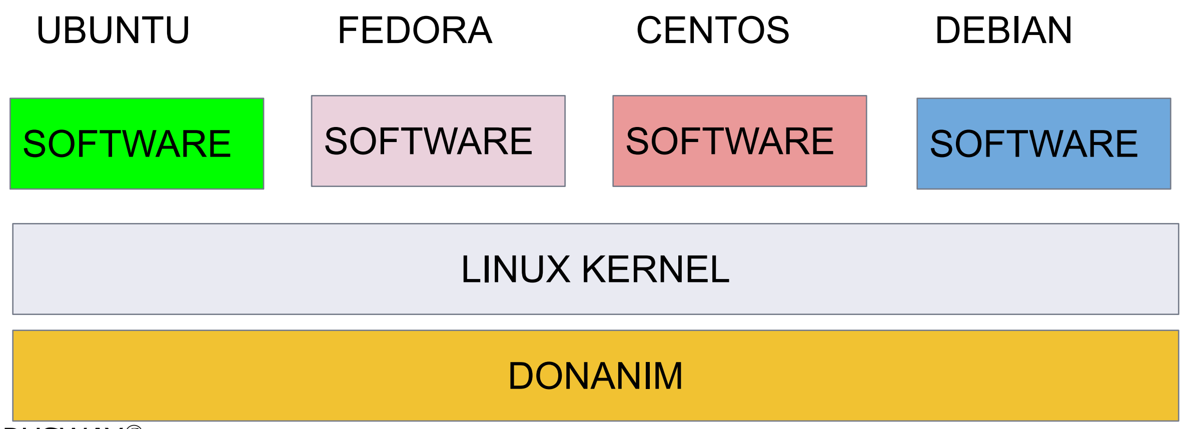
You can find me at:

- james@clarusway.com

Students, write your response!

Pear Deck Interactive Slide
Do not remove this bar

What is Linux Distributions


UBUNTU
FEDORA
CENTOS
DEBIAN
SOFTWARE
SOFTWARE
SOFTWARE
SOFTWARE
LINUX KERNEL
DONANIM