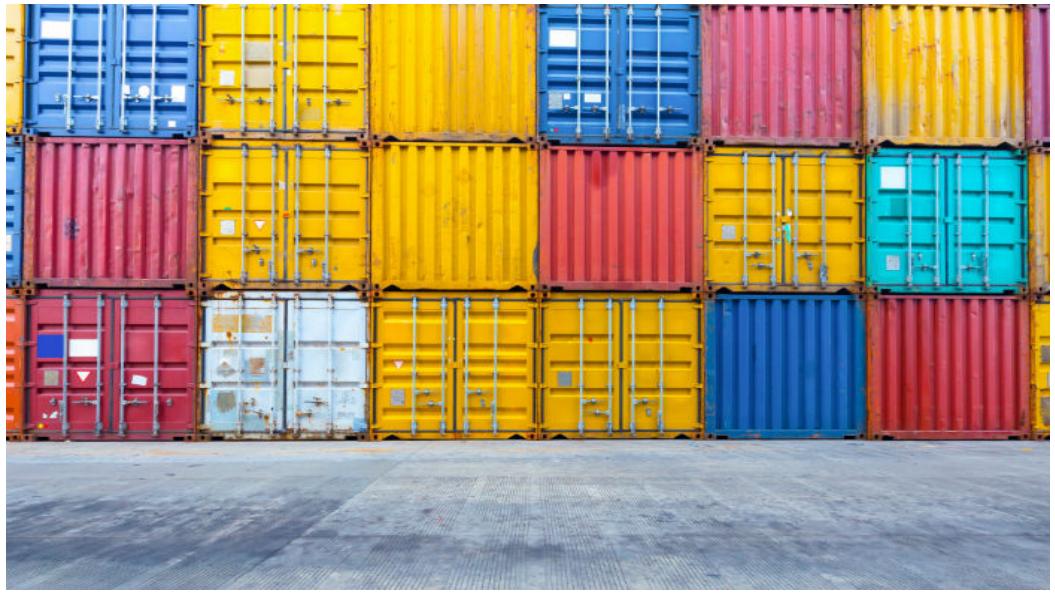
Virtualization and Containerization

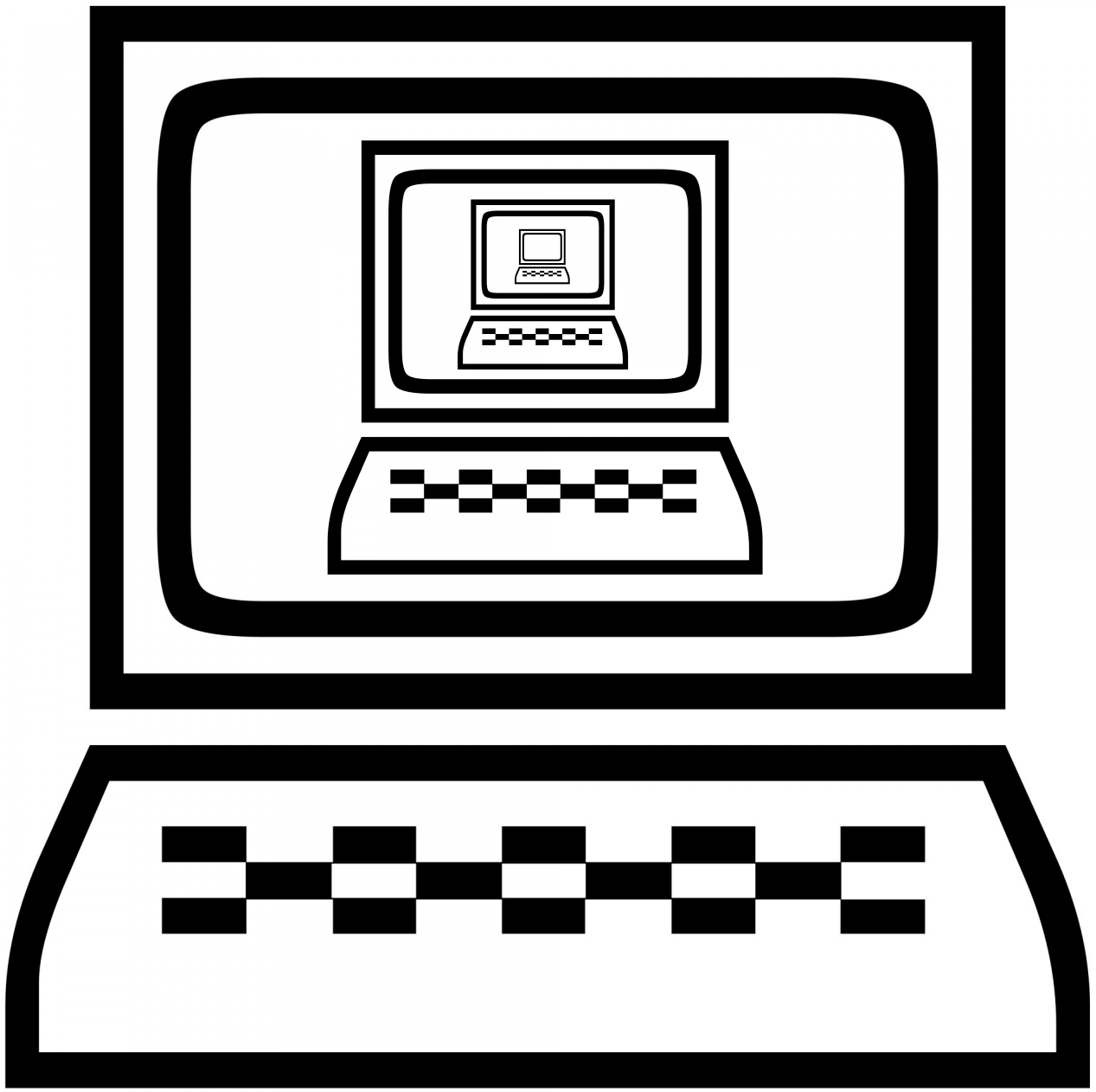


At a **high level**, how do computers work?

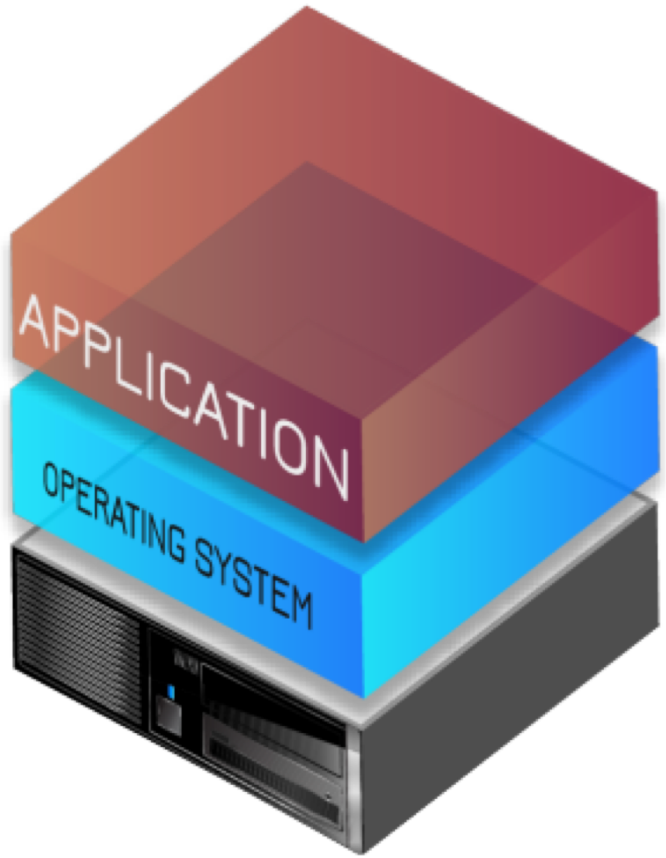
Traditional Computers

Linux

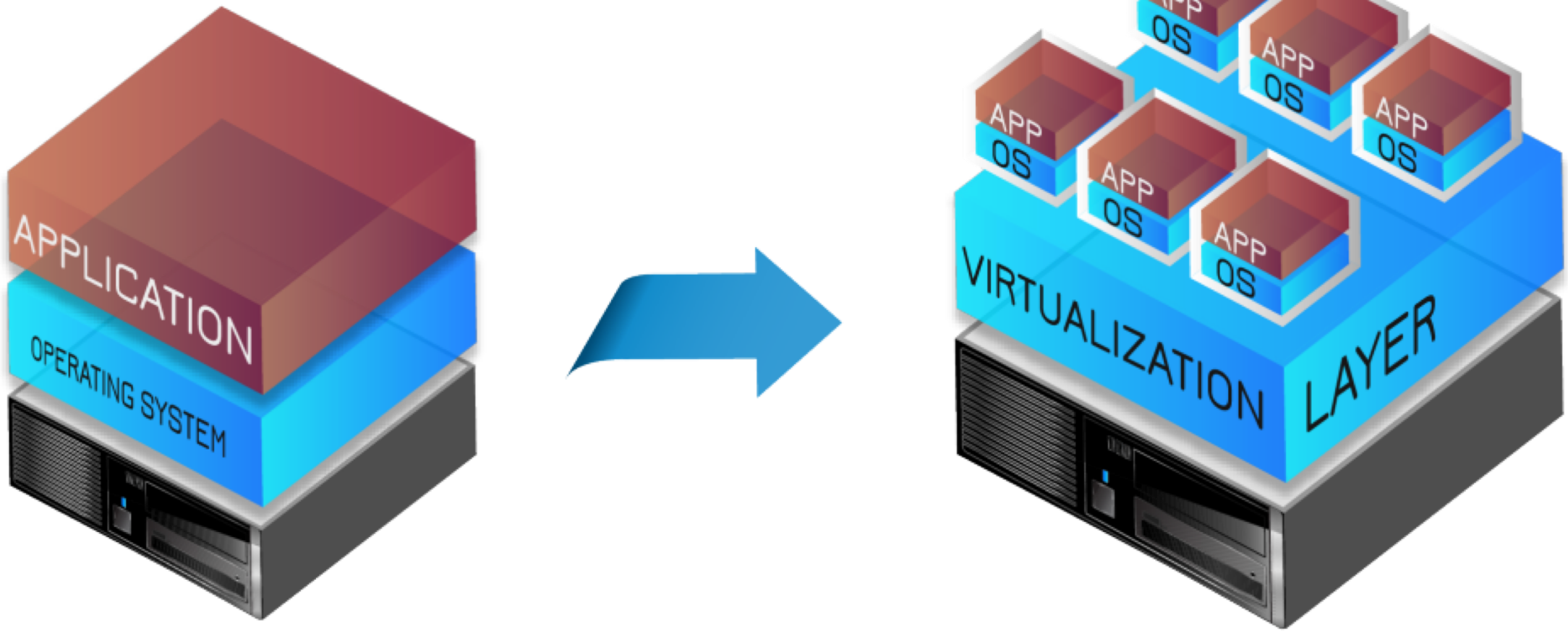
Take a guess at what Virtualization and Containerization are…



Virtualization

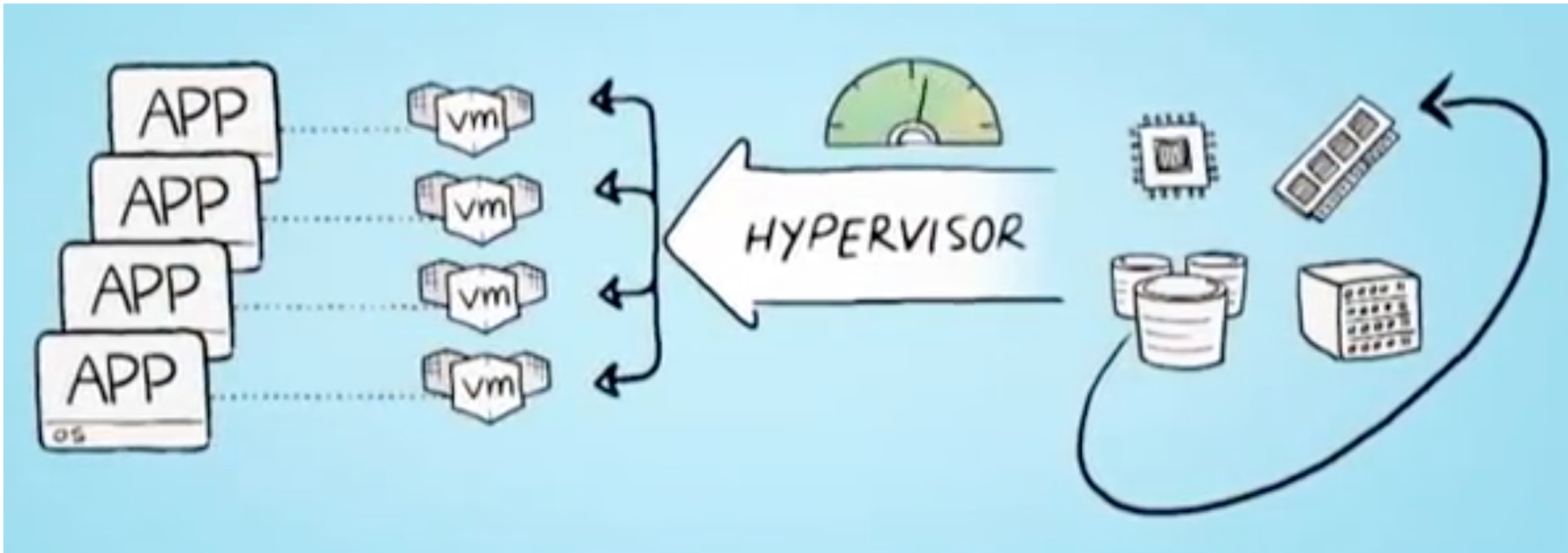


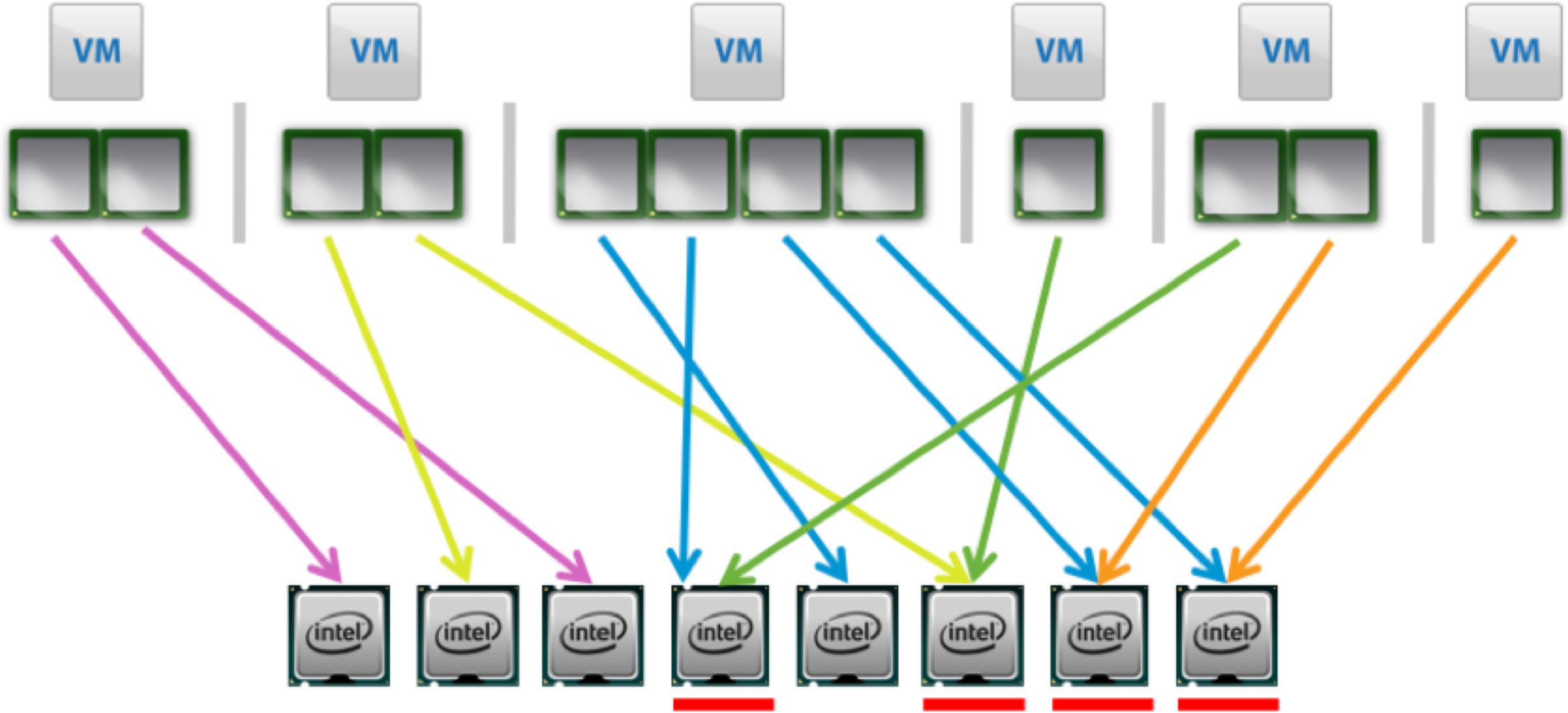
Traditional Architecture

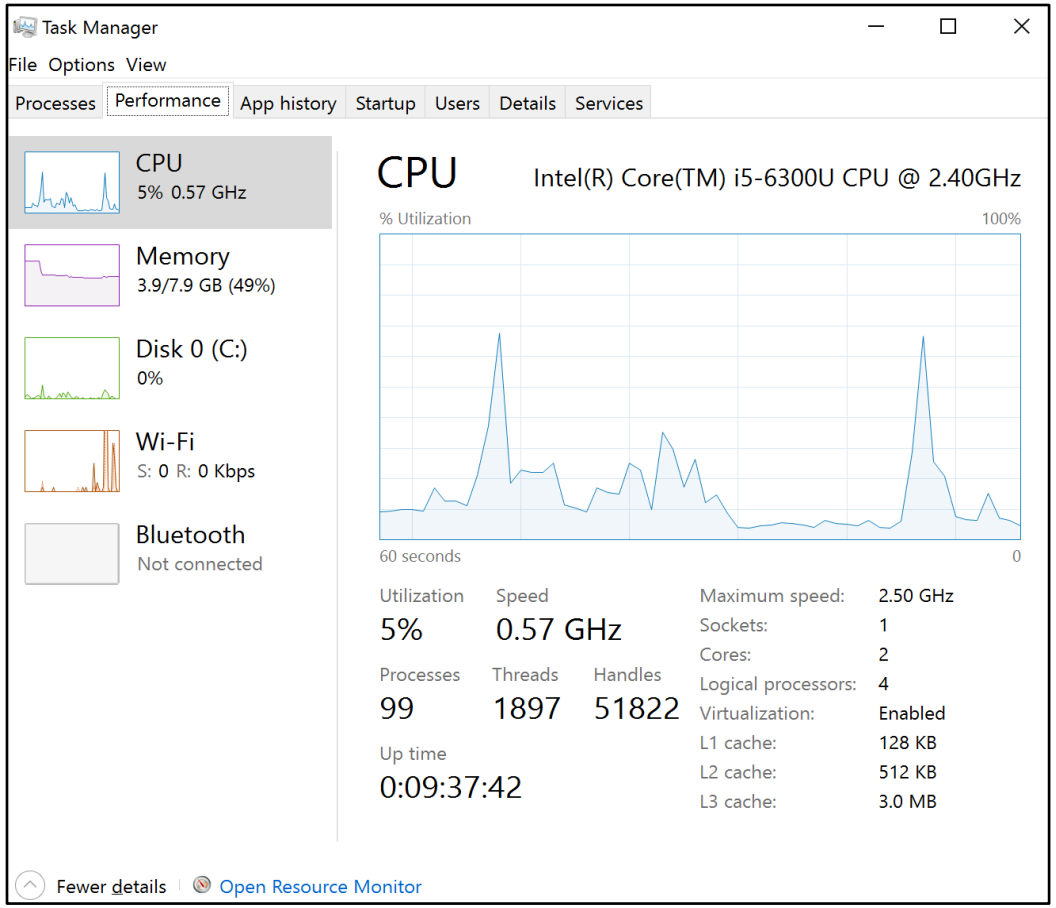


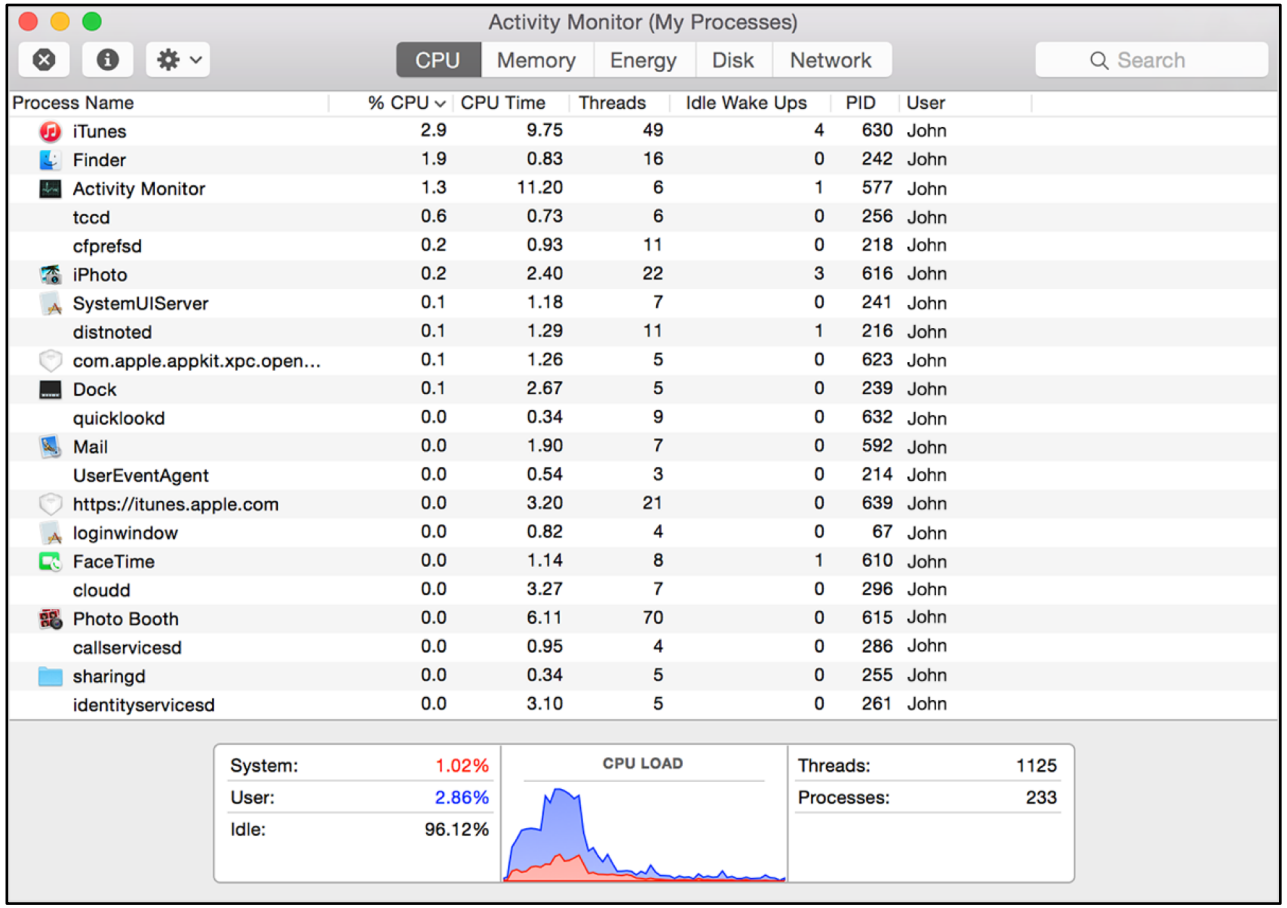
Traditional Architecture

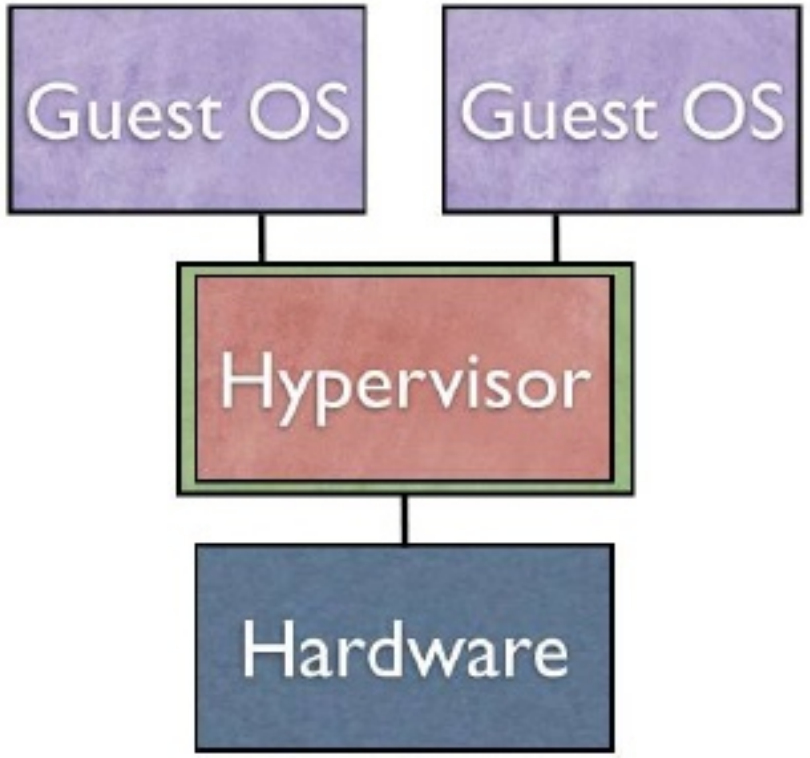
Virtual Architecture


APP
APP
APP
APP
OS
vm
vm
vm
vm
HYPERVISOR

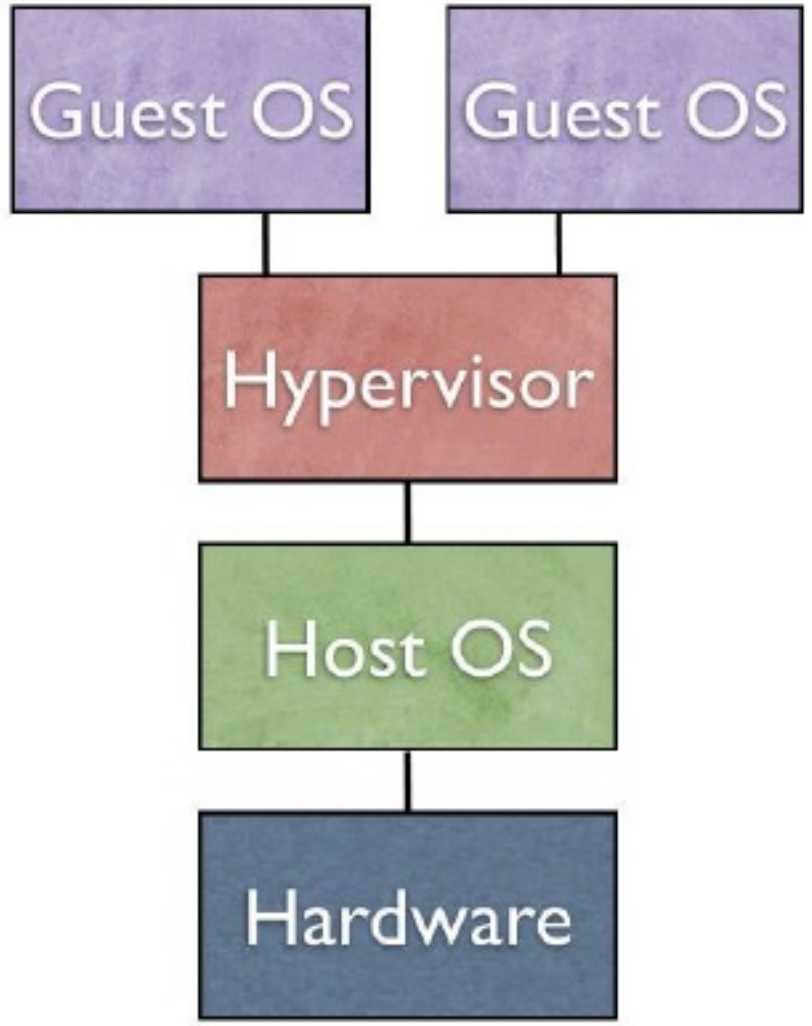

VM
VM
VM
VM
VM
VM
intel
intel
intel
intel
intel
intel
intel
intel


Task Manager
File Options View
Processes Performance App history Startup Users Details Services
CPU
5% 0.57 GHz
Memory
3.9/7.9 GB (49%)
Disk 0 (C:)
0%
Wi-Fi
S: 0 R: 0 Kbps
Bluetooth
Not connected
CPU
Intel(R) Core(TM) i5-6300U CPU @ 2.40GHz
% Utilization
100%
60 seconds
0
Utilization
5%
Speed
0.57 GHz
Processes
99
Threads
1897
Handles
51822
Up time
0:09:37:42
Maximum speed: 2.50 GHz
Sockets: 1
Cores: 2
Logical processors: 4
Virtualization: Enabled
L1 cache: 128 KB
L2 cache: 512 KB
L3 cache: 3.0 MB
Fewer details
Open Resource Monitor


Activity Monitor (My Processes)
CPU Memory Energy Disk Network
Search
Process Name % CPU CPU Time Threads Idle Wake Ups PID User
iTunes 2.9 9.75 49 4 630 John
Finder 1.9 0.83 16 0 242 John
Activity Monitor 1.3 11.20 6 1 577 John
tccd 0.6 0.73 6 0 256 John
cfprefsd 0.2 0.93 11 0 218 John
iPhoto 0.2 2.40 22 3 616 John
SystemUIServer 0.1 1.18 7 0 241 John
distnoted 0.1 1.29 11 1 216 John
com.apple.appkit.xpc.open... 0.1 1.26 5 0 623 John
Dock 0.1 2.67 5 0 239 John
quicklookd 0.0 0.34 9 0 632 John
Mail 0.0 1.90 7 0 592 John
UserEventAgent 0.0 0.54 3 0 214 John
https://itunes.apple.com 0.0 3.20 21 0 639 John
loginwindow 0.0 0.82 4 0 67 John
FaceTime 0.0 1.14 8 1 610 John
cloudd 0.0 3.27 7 0 296 John
Photo Booth 0.0 6.11 70 0 615 John
callservicesd 0.0 0.95 4 0 286 John
sharingd 0.0 0.34 5 0 255 John
identityservicesd 0.0 3.10 5 0 261 John
System: 1.02%
User: 2.86%
Idle: 96.12%
CPU LOAD
Threads: 1125
Processes: 233



Type 1 (Hardware Level)



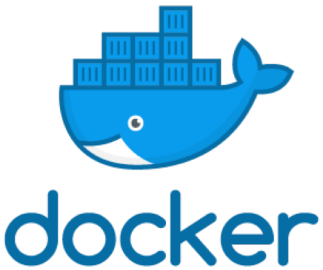
Type 2 (OS Level)

Characteristics of Virtual Machines

- **Partitioning**
  - Run multiple operating systems on one physical machine.
  - Divide system resources between virtual machines.
- **Isolation**
  - Provide fault and security isolation at the hardware level.
  - Preserve performance with advanced resource controls.
- **Encapsulation**
  - Save the entire state of a virtual machine to files.
  - Move and copy virtual machines as easily as moving and copying files.
- **Hardware Independence**
  - Provision or migrate any virtual machine to any physical server.

Benefits of Virtualization

- Higher performance at a lower cost due to resource optimization
- Better security as you can prevent operating systems from communicating to one another
- Reduced capital and operating costs
- Faster provisioning of applications and resources

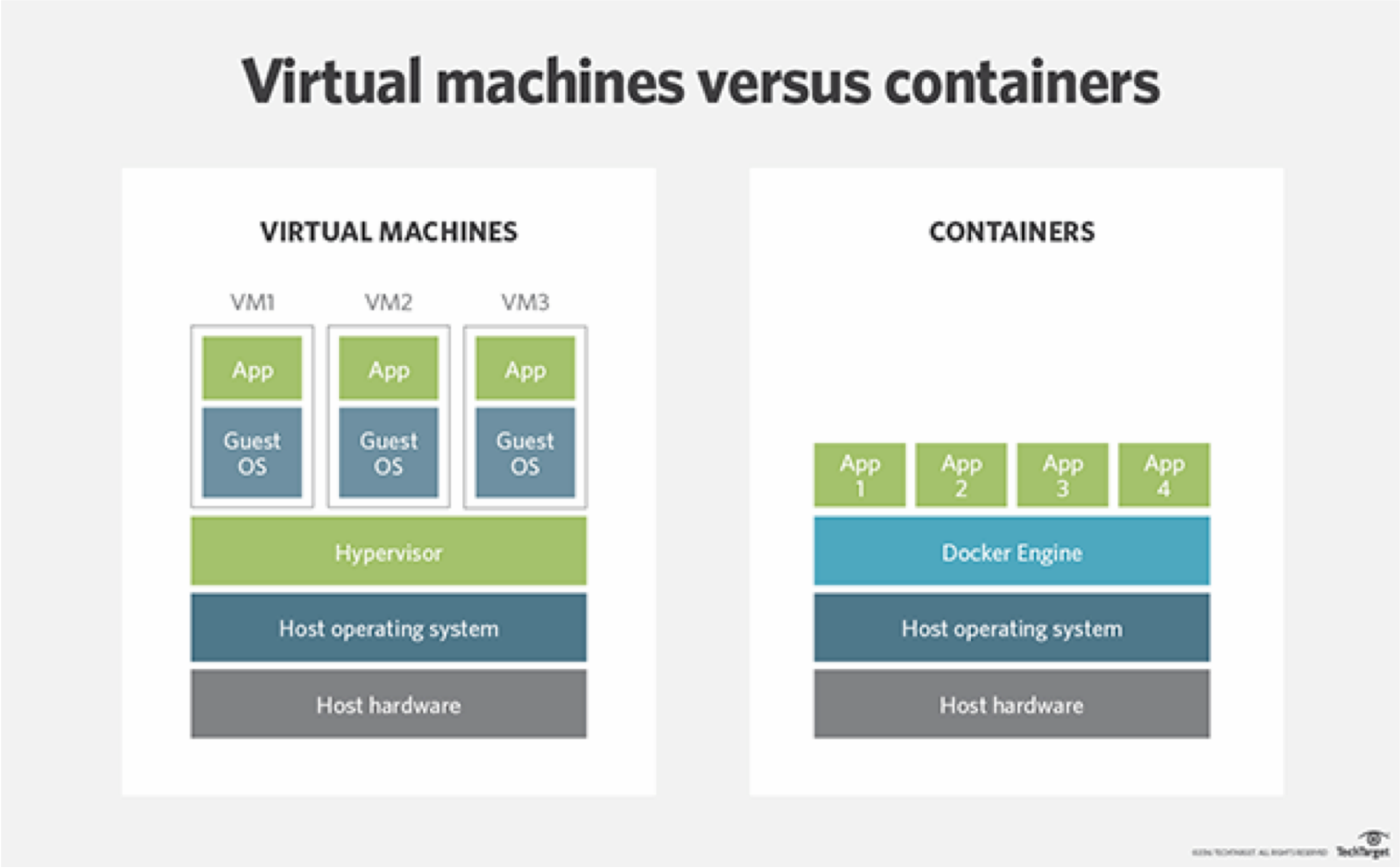


Containers!

What are Containers?

- Lightweight, portable software packages
  - Only contain the software, libraries, and other dependencies they require to run so you can put them directly on the host operating system.
- Smaller version of Virtual Machines, only have the bare necessities
  - 20 MB vs 20 GB
- Operating system kernel via API calls with other containers on a host
- Many Variations
  - Most Popular = Docker

Containers vs. Virtual Machines



Why Use Containers?

- Performance
  - Without any hypervisor, much more stable without the need of more resources
- Reliability
  - Predictable apps created within containers
  - Distribution ensures anything downloaded works the exact same within container
- App Isolation
  - Developers get exactly what they need, no more no less, only dependencies needed are downloaded
    - Can alter changes within the container but not on the host OS
  - Allows for greater security - Split resources by container
- Availability
  - Can be downloaded anywhere, anytime and on ANY operating system – Very Portable
    - Will work the same way across platforms (Mac, Linux, Windows)

Issues with Containerization

- No isolation from host OS (Docker)
  - Root access required
  - Once inside container, have root access
- Container Malware
  - Can be distributed much more easily with containers
- Denial of Service Attacks
  - Multiple containers share same Kernel resources
  - If you can grab kernel resources, can starve all containers on system

What's the Difference??

	Containers	Virtual Machines
Consistent Runtime Environment	X	X
Application Sandboxing	X	X
Resource Allocation	X	X
Small Size on Disk	X	
Low Overhead	X	

Can I use a virtual machine?

Server: cdr-vcenter1.cse.buffalo.edu

Username: ad\UBIT

Password: UBIT Password