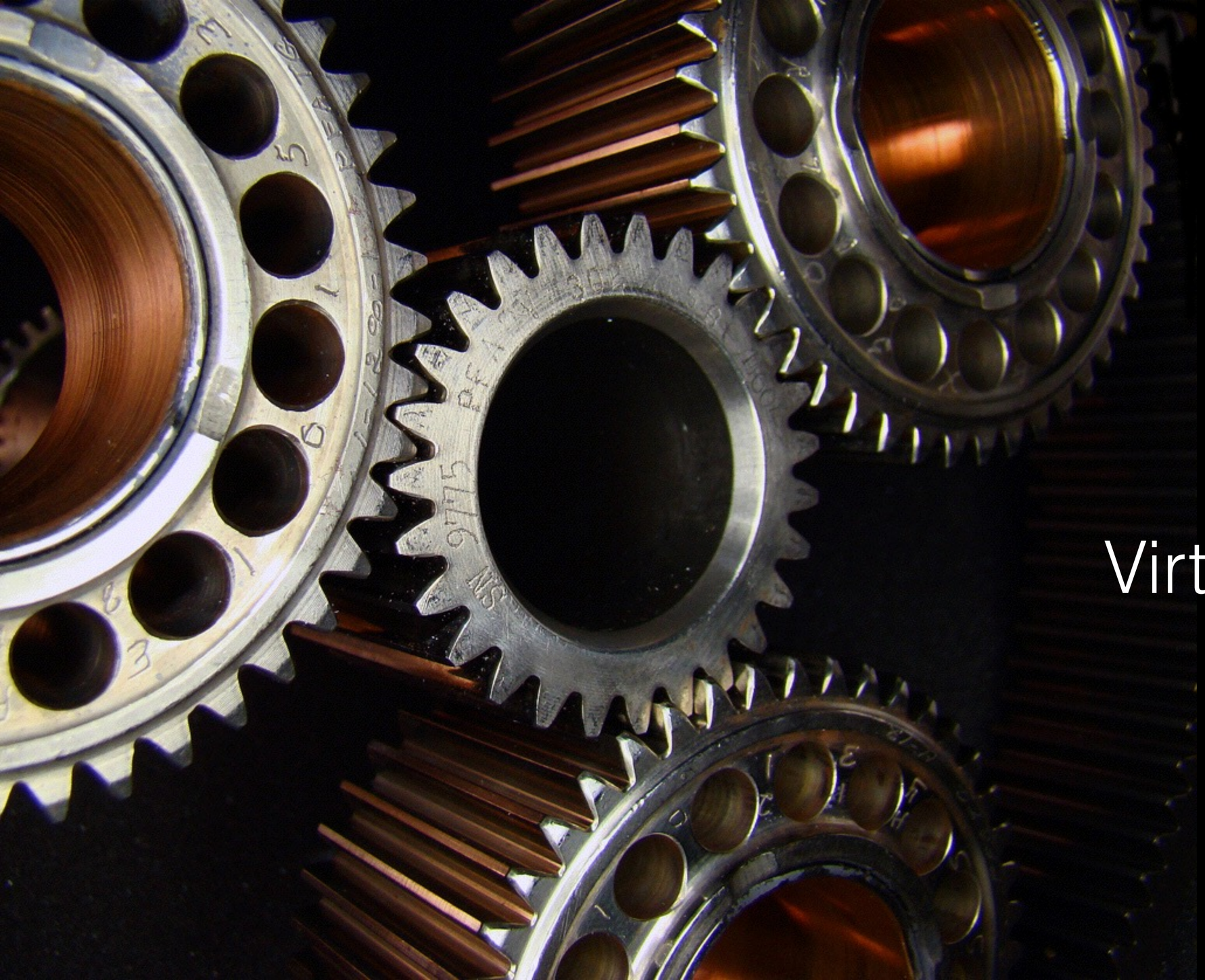


Virtualization

Mark Nelson

October 10, 2016

Virtualawhat?

Ancient History

- Early hardware ran dedicated programs — for computing ballistics or running payroll
- Early operating systems only allowed one program to run at a time
- OS' that supported multiple processes was a primitive incarnation of virtualization
- In the late 90's, Intel CPUs were fast enough for researchers to try and virtualize an entire OS
  - Xen and VMWare started with *emulation*…
  - But needed just a little hardware support, so they collaborated with Intel

Virtualization Defined

It's basically a computer inside a computer…

A "host" computer emulates hardware components…

To create an illusion for "guests" that they are running on their own hardware.

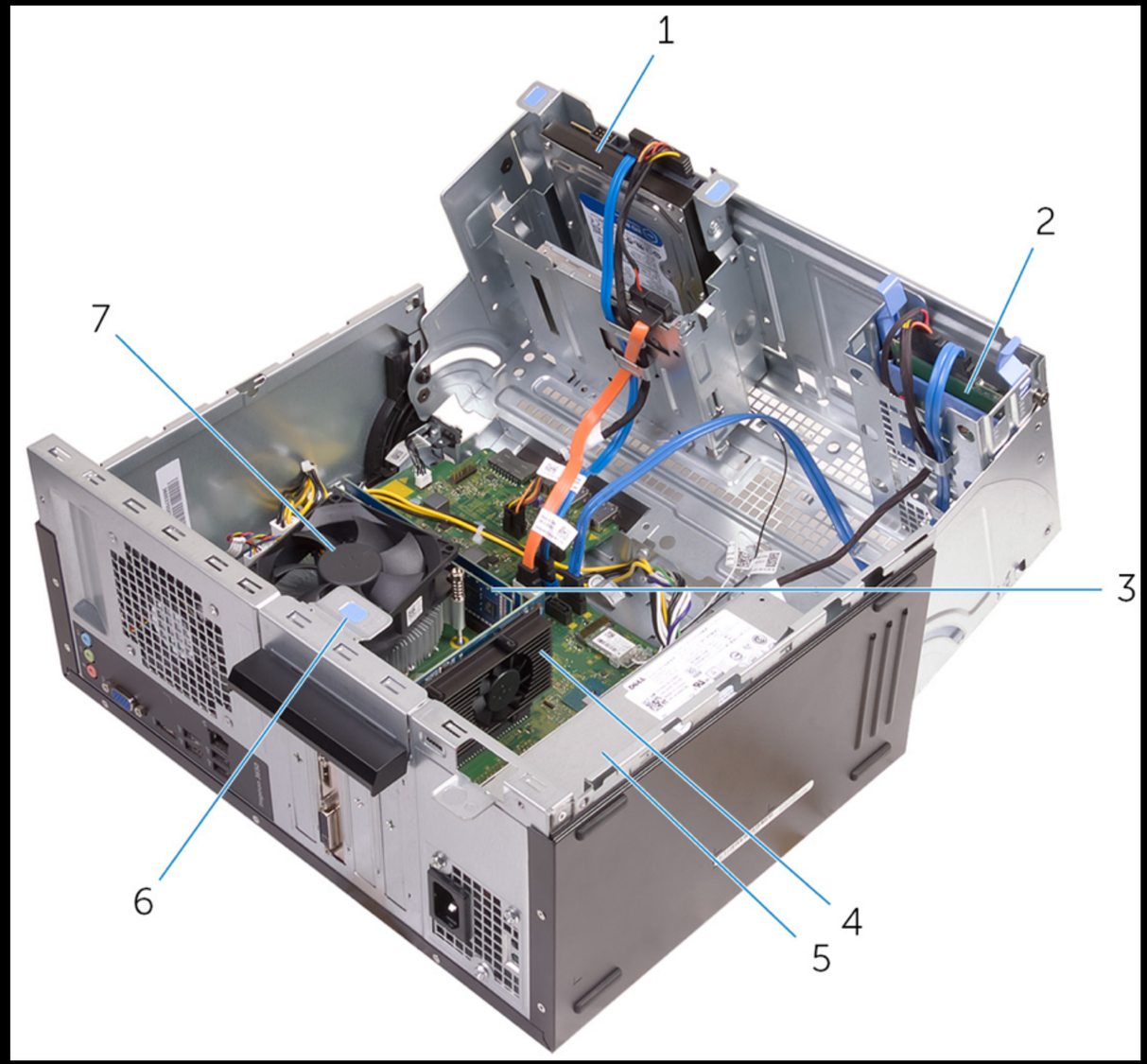
It allows multiple virtual machines, with heterogeneous operating systems to run in isolation, side-by-side on the same physical machine.

Virtualization is an abstraction layer that decouples the physical hardware from the operating system to deliver greater IT resource utilization and flexibility.

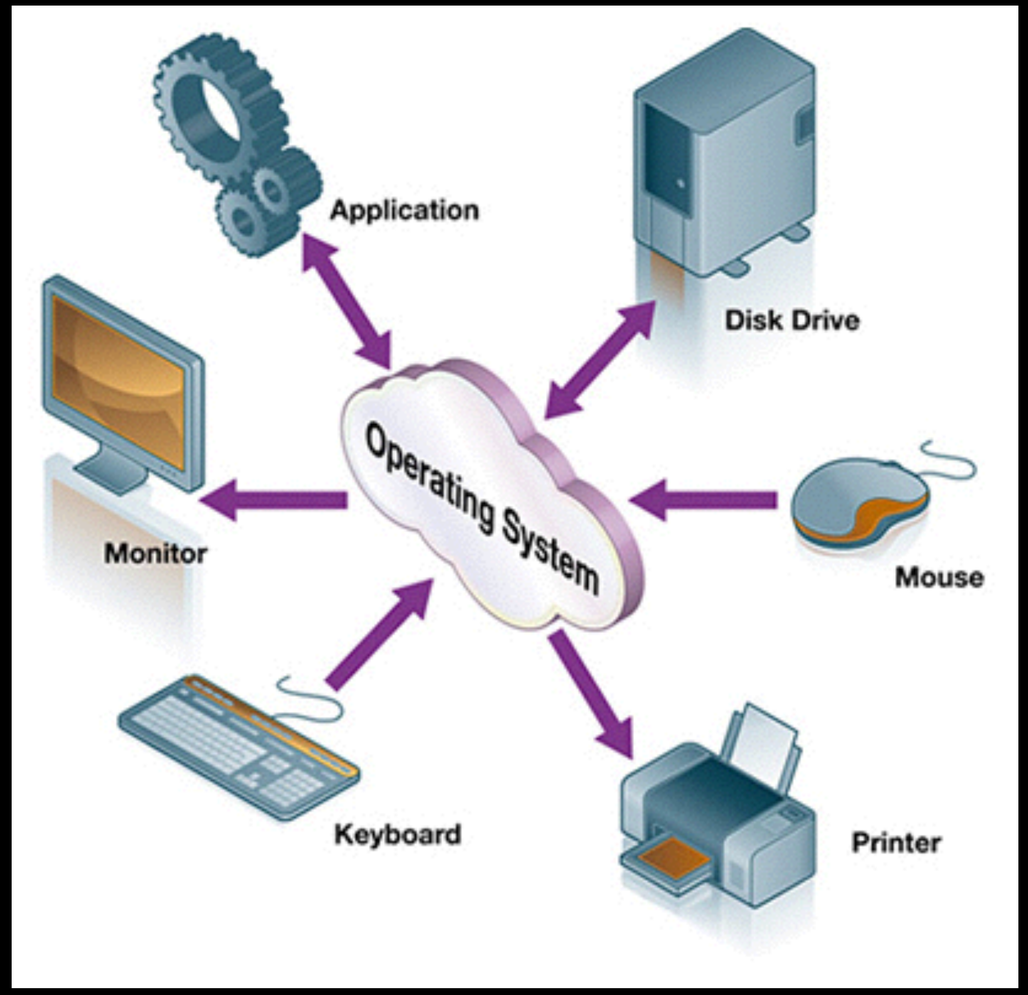
Demo

What's Inside…

- Hard Drive
- CPU
- Memory
- Power Supply
- CD/DVD
- Video Card
- USB Ports
- …and other stuff



What is an Operating System?

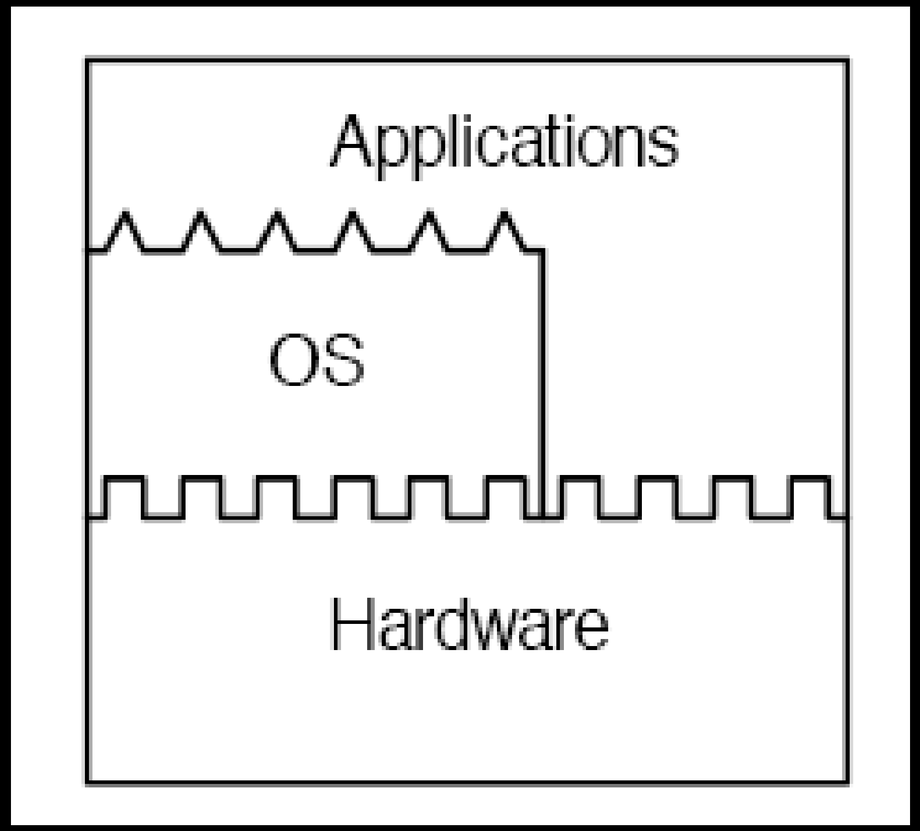


- An Operating System (OS) is a group of programs that:
  1. Allocates computing system resources
  2. Provides an interface between the computer hardware and the programmer

- It simplifies programming, debugging and maintenance of application programs

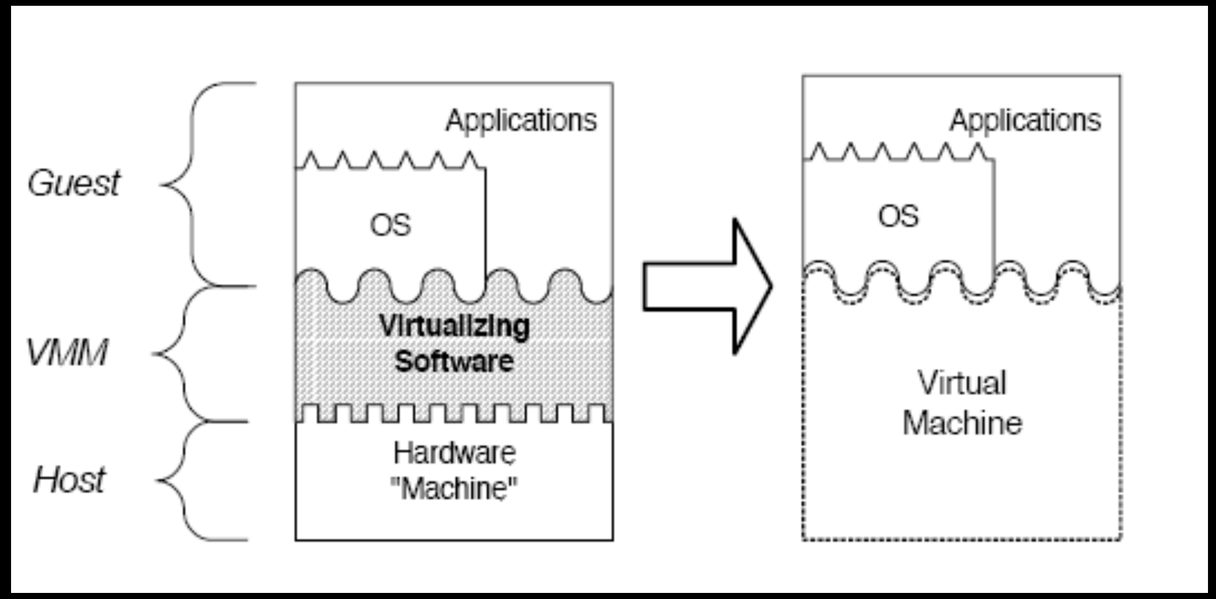
…in other words…

- The OS talks to Hardware…
- …and the applications (usually) talk to the OS…



But what if…

- We could write a program to trick an Operating System… into thinking it's talking to real hardware



Guest Hardware	A supervising program could emulate it by…
Hard Disk, DVD	Each HDD could be a big file
Memory	Allocate a large chunk of memory and give it to the guest
Video Card	Render the screen as an application window
USB Ports	Maybe we could selectively pass them through
CPU	Who cares? Just run it on the real CPU

Demo

The Big Picture

- At this point, I want to make sure that you “Get it”… because it’s so simple sometimes people think it’s harder than it really is

- …that said, once you “Get it”… the implications of virtualization is what is hard to get your head wrapped around

A dirty secret

Most hardware is underutilized

Virtualization fixes that

Basic Benefits of Virtualization

- Partitions hardware so you can run multiple OS' or Applications
  - If one guest crashes, it doesn't take down the whole shebang
- Better utilization of hardware: Smoothes out the load
- The OS sees a consistent, predictable set of hardware rather than whatever you really have
- Programs have APIs

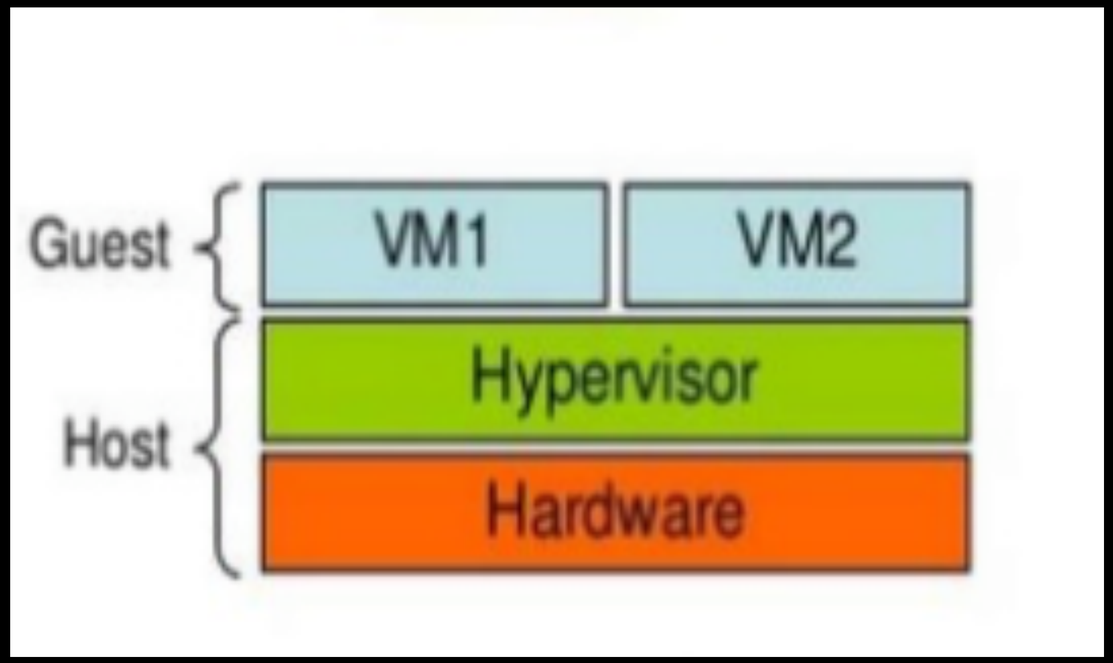
Types of Virtualization

The fun of creation: There's more than one way to do it…

Type	What It Is	Who Does It
Hardware-level Hypervisors	"Monitors" are built into the hardware/BIOS to partition OS's — The OS's run on bare metal	HP, IBM and Oracle/Sun
OS-level Hypervisors	The OS is the Hypervisor — it's the only thing the system does	VMWare ESX, Citrix Xen
Process-level Hypervisors	The Hypervisor is an application. The host runs regular programs and guests	Hyper-V, RHL KVM, VMWare Workstation, Parallels, Oracle Virtual Box
Language-level VMs (not really Hypervisors)	Interpreted code running in their own "environments"	Java, .NET & PHP run in JVMs or CLRs
Emulators	No hardware support	Bochs, QEMU
Containers	Not really virtualization, but a close partnership between hosts and guests to segregate applications	Docker, Heroku

A Picture

OS Level



Specialization Rocks

It ends up that there are so many benefits that the virtualization industry has specialized

Type	Where	What, How & Why
Server Virtualization	In the datacenter…	Buy a big box, install a Hypervisor and provision a bunch of OS's + Apps
Desktop Virtualization	On your rig…	So you can have Windows, OS X and Linux 24x7
VDI	At work…	So, there's two ways to provision an office, a bunch of workstations … or … A bunch of less-expensive computers and your "work PC" is actually hosted in a datacenter
Containers	For your apps…	Write and deploy to containers, not Operating Systems
Cloud Computing	Well, in the cloud, of course…	If you don't want the hassle of paying the electric bill. Amazon, Backspace & Azure

Credits & References

- *Virtual Machines* by Sai Siddhartha Kumar Dantu
- *An Overview of Virtual Machine Architectures* by Ross Rosemark
- *Virtualization* by Sam Shamsuddin

“Cloud is about how you do computing, not where you do computing”

–Paul Maritz, CEO of VMWare

“If you’re bad at IT, you’re going to be really bad at virtualization”

–Steve Chambers

“You don’t really understand something until you can explain it to your Grandmother”

–Albert Einstein

Benefits

Management	Business	Programming
Snapshots Cloning Metering Live Migration Scalable API	Diaster Recovery Efficiency Security Flexibility Training	Consistent Environment Dev, Test, Prod Segregation