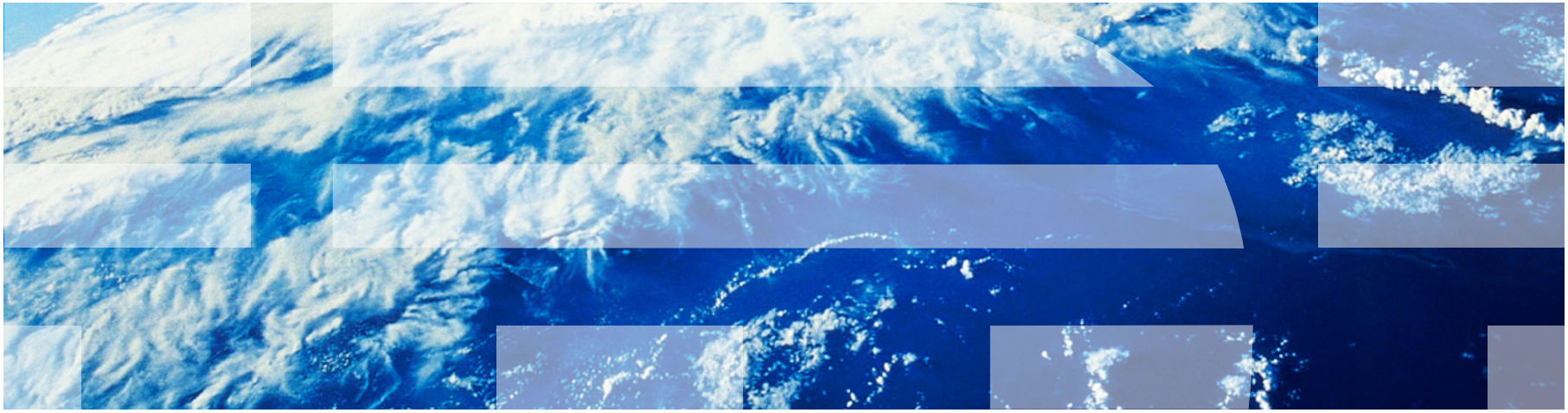
Lecture 5: Virtualization Basics

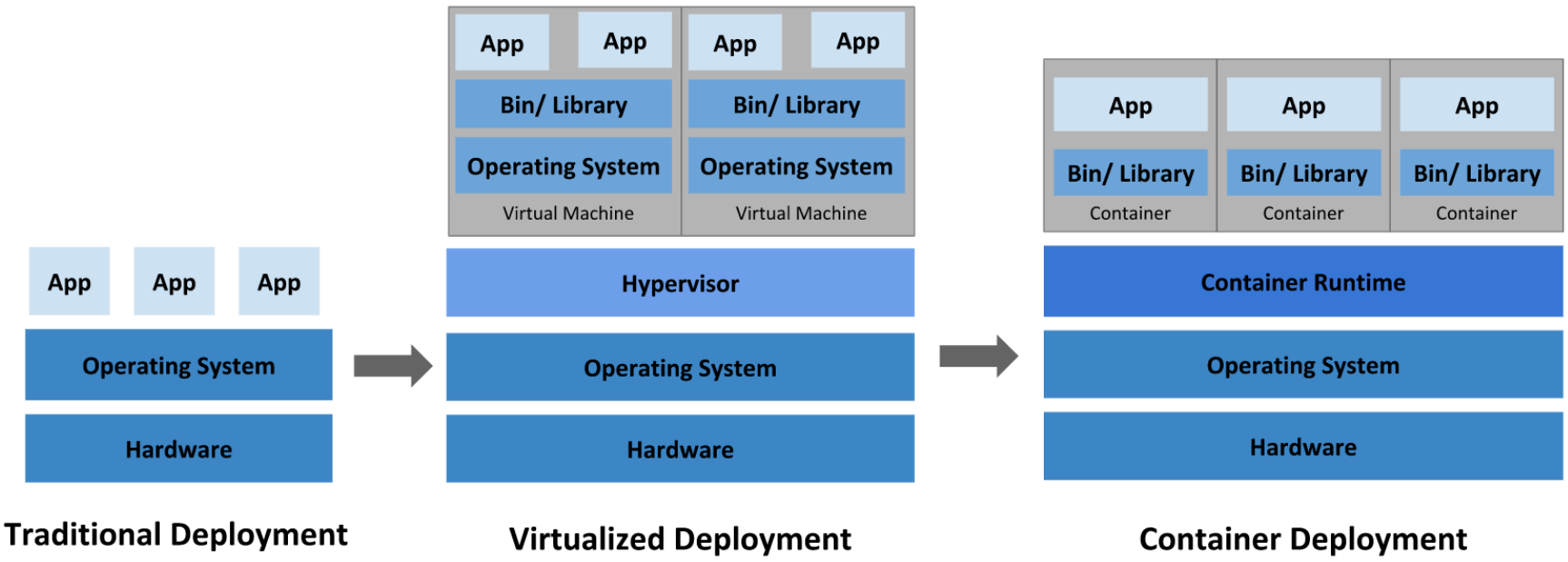
Sambit Sahu, IBM Research



Reading Materials

- VMWare virtualization concepts: https://www.youtube.com/watch?v=EvXn2QiL3gs
- Xen: Art of Virtualization: https://www.cl.cam.ac.uk/research/srg/netos/papers/2003-xensosp.pdf
- RedHat Virtualization: https://www.redhat.com/en/topics/virtualization
- https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/

Physical Machines/ Virtual Machines/ Containers

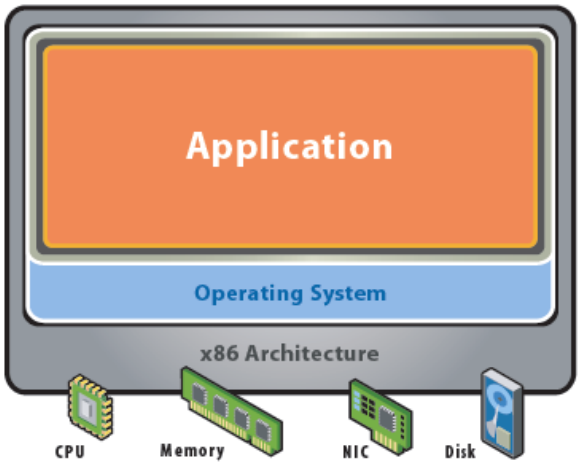


Virtualization

- Actually an old concept since 1960's invented by IBM, available on mainframes
- Virtualization is a technology
- Cloud computing is a service that is enabled by virtualization technology

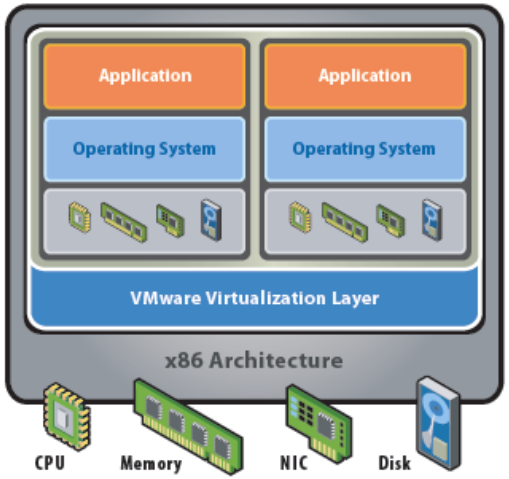
What is virtualization?

- Creation of virtual version of something such as a server, network, storage, etc.
- Separation of a resource or request for a service from the underlying physical delivery of that service.



Before Virtualization:

- Single OS image per machine
- Software and hardware tightly coupled
- Running multiple applications on same machine often creates conflict
- Underutilized resources
- Inflexible and costly infrastructure



After Virtualization:

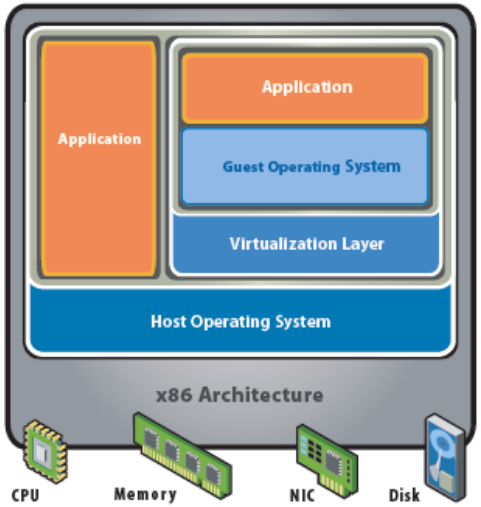
- Hardware-independence of operating system and applications
- Virtual machines can be provisioned to any system
- Can manage OS and application as a single unit by encapsulating them into virtual machines

Who provides (server) virtualization software products?

- Commodity x86 CPUs
  - VMWare ESXi Hypervisor / VSphere Platform
  - Linux KVM (Kernel Virtual Machine) Hypervisor
  - Citrix Xen Hypervisor
  - Microsoft Hyper-V Hypervisor
  - Sun/Oracle VirtualBox
  - etc.
- Other CPUs
  - IBM PowerVM Hypervisor
  - IBM System z/VM Hypervisor

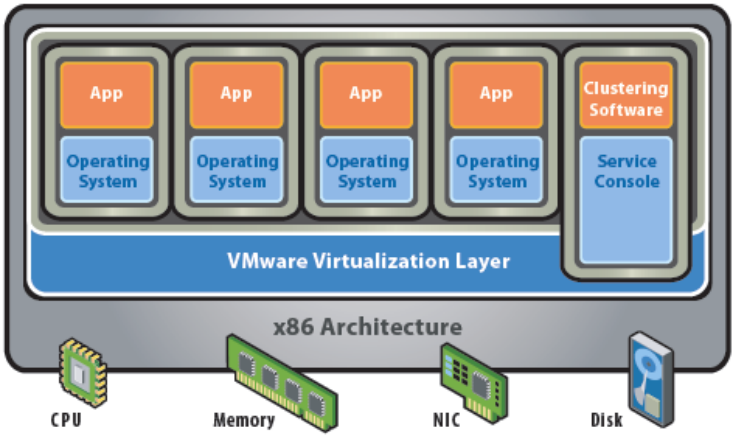
Hosted (i.e., VMWare Player/Fusion) vs. Hypervisor (i.e., VMWare ESXi)

- Hypervisor == VMM (Virtual Machine Monitor)
- Modern hypervisors take advantage of modern CPUs, with special virtualization instructions that provide better performance, i.e. Intel's "VT" and AMD's "Pacifica"



Hosted Architecture

- Installs and runs as an application
- Relies on host OS for device support



Bare-Metal (Hypervisor) Architecture

- Lean virtualization-centric kernel
- Service Console for agents and helper

Advantages of virtualization

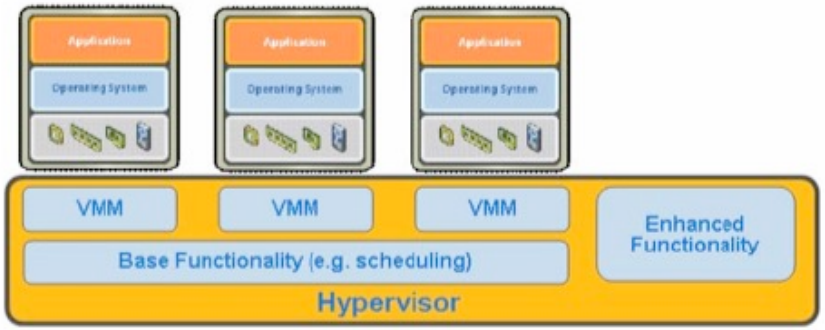
- Run ***multiple*** operating systems on a single physical system and share the underlying hardware resources – known as ***partitioning.***
- A virtual infrastructure gives administrators the advantage of ***managing pooled resources*** across the enterprise.
- Enables ***Server Consolidation and Containment*** – Eliminating 'server sprawl' via deployment of systems as virtual machines (VMs) that can run safely and move transparently across shared hardware, and increase server utilization rates from 5-15% to 60-80%. Can also help save power (smaller energy bills via consolidation).
- ***Test and Development Optimization*** – Rapidly provisioning test and development servers by reusing pre-configured systems, enhancing developer collaboration and standardizing development environments.
- ***Improved Business Continuity*** – Reducing the cost and complexity of business continuity (high availability and disaster recovery solutions) by encapsulating entire systems into single files that can be replicated and restored on any target server, thus minimizing downtime.

Types of Virtualization

- Hardware emulation
  - Most complex: a hardware VM is created for each instance
- Full Virtualization
  - Uses hypervisor to share underlying hardware across guest VMs
  - Mediates between guest OS and underlying h/w
- Paravirtualization
  - Differs from full virtualization in that integrates virtualization handling code into OS – thus the guest OS code is modified
- Operating system level virtualization
  - Virtualizes server on top of operating systems
  - Single OS that isolates the servers

Hypervisor and VMM

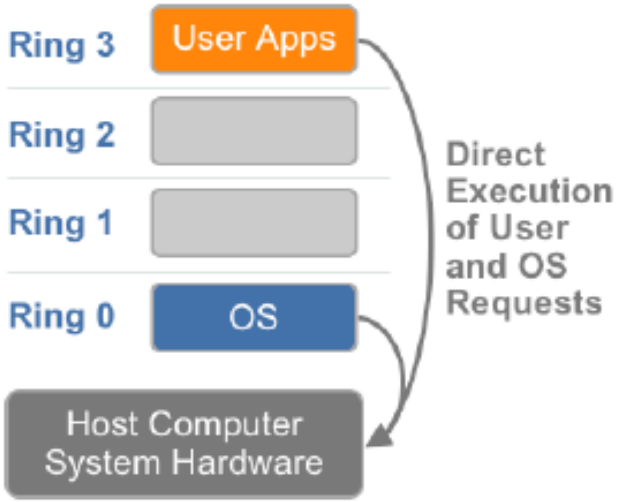
- Hypervisor runs on bare metal machine
- Functionality/role of hypervisor is dependent o type of virtualization



Hypervisor manages virtual machine monitors

- So what is required in supporting virtualization, i.e., running multiple OS instances on a single machine?
  - OS typically has all the privilege, ring 0-3
  - Need to somehow not allow all the OS instances to run at ring 0, but still be able to function as OS

- Solution
  - Hypervisor runs at ring 0
  - OS runs at higher layer than ring 0, but lower than user applications
  - OS level instructions that required ring 0 privilege → need to be now run by hypervisor instead!



Three types of virtualization (for CPU)

- Depending on how hypervisor handles the critical instructions from OS (ring 0), there are different virtualization methods
  - Full virtualization using binary translation
  - OS assisted virtualization or Paravirtualization
  - Hardware assisted virtualization

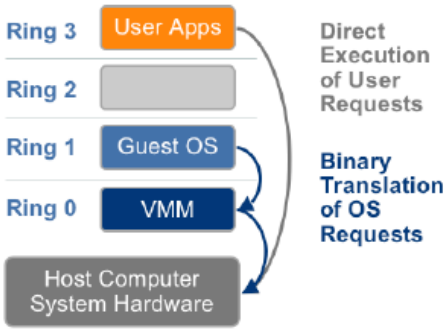


Figure 5 – The binary translation approach to x86 virtualization

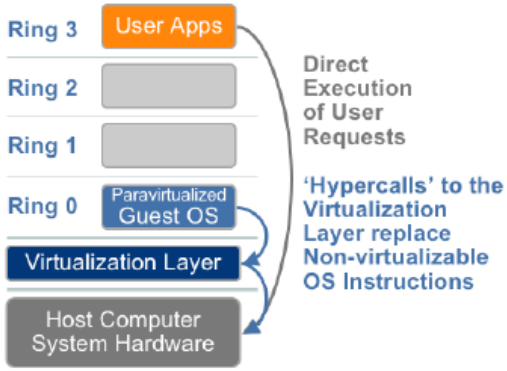


Figure 6 – The Paravirtualization approach to x86 Virtualization

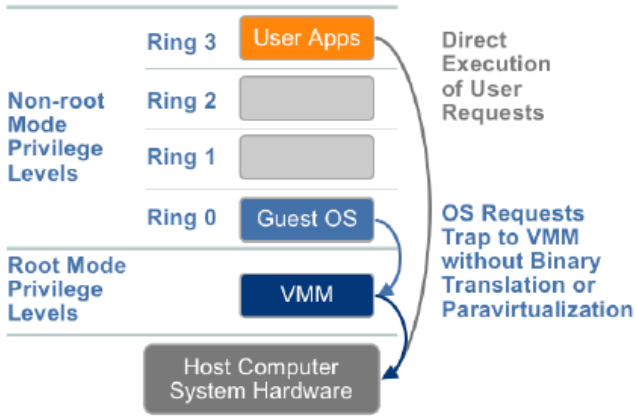


Figure 7 – The hardware assist approach to x86 virtualization

Full Virtualization

Para Virtualization

H/W Assisted Virtualization

Memory Virtualization

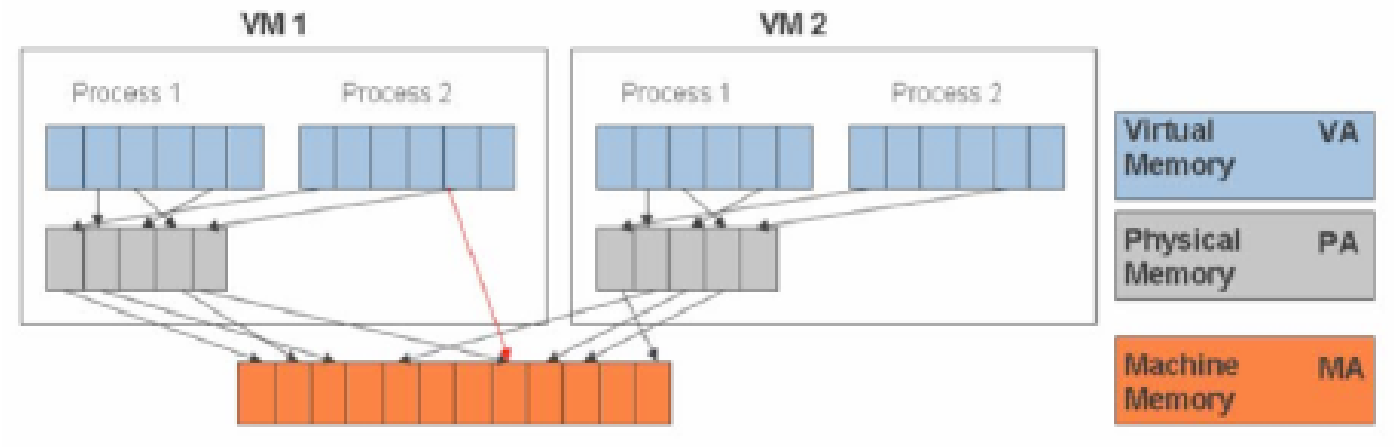


Figure 8 – Memory Virtualization

- Memory virtualization requires further virtualization of OS level virtual memory
  - Another level of MMU virtualization that maps multiple MMU into physical memory

Device and I/O Virtualization

- Requires managing the routing of I/O requests between virtual devices and shared physical hardware
- Example: Virtual Network Interface and switches
- Virtual devices emulate the physical devices

Comparison of Three Virtualization Methods

	Full Virtualization with Binary Translation	Hardware Assisted Virtualization	OS Assisted Virtualization / Paravirtualization
Technique	Binary Translation and Direct Execution	Exit to Root Mode on Privileged Instructions	Hypercalls
Guest Modification / Compatibility	Unmodified Guest OS Excellent compatibility	Unmodified Guest OS Excellent compatibility	Guest OS codified to issue Hypercalls so it can't run on Native Hardware or other Hypervisors Poor compatibility; Not available on Windows OSes
Performance	Good	Fair Current performance lags Binary Translation virtualization on various workloads but will improve over time	Better in certain cases
Used By	VMware, Microsoft, Parallels	VMware, Microsoft, Parallels, Xen	VMware, Xen
Guest OS Hypervisor Independent?	Yes	Yes	XenLinux runs only on Xen Hypervisor VMI-Linux is Hypervisor agnostic

Figure 10 – Summary comparison of x86 processor virtualization techniques

Iterative Memory Copy for Live Migration

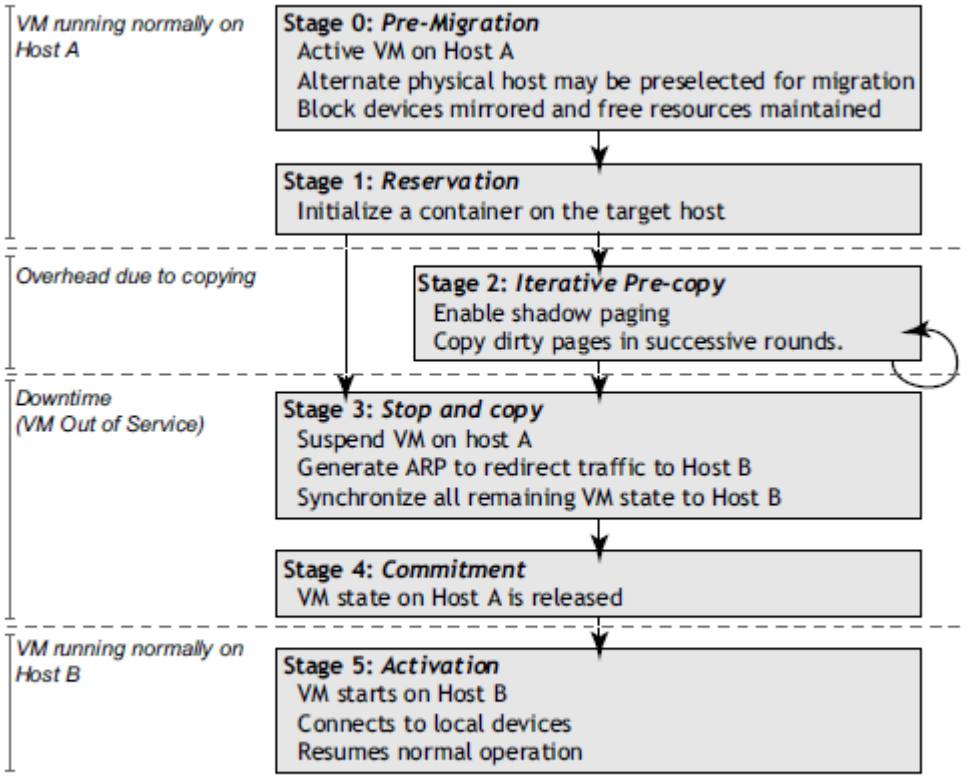
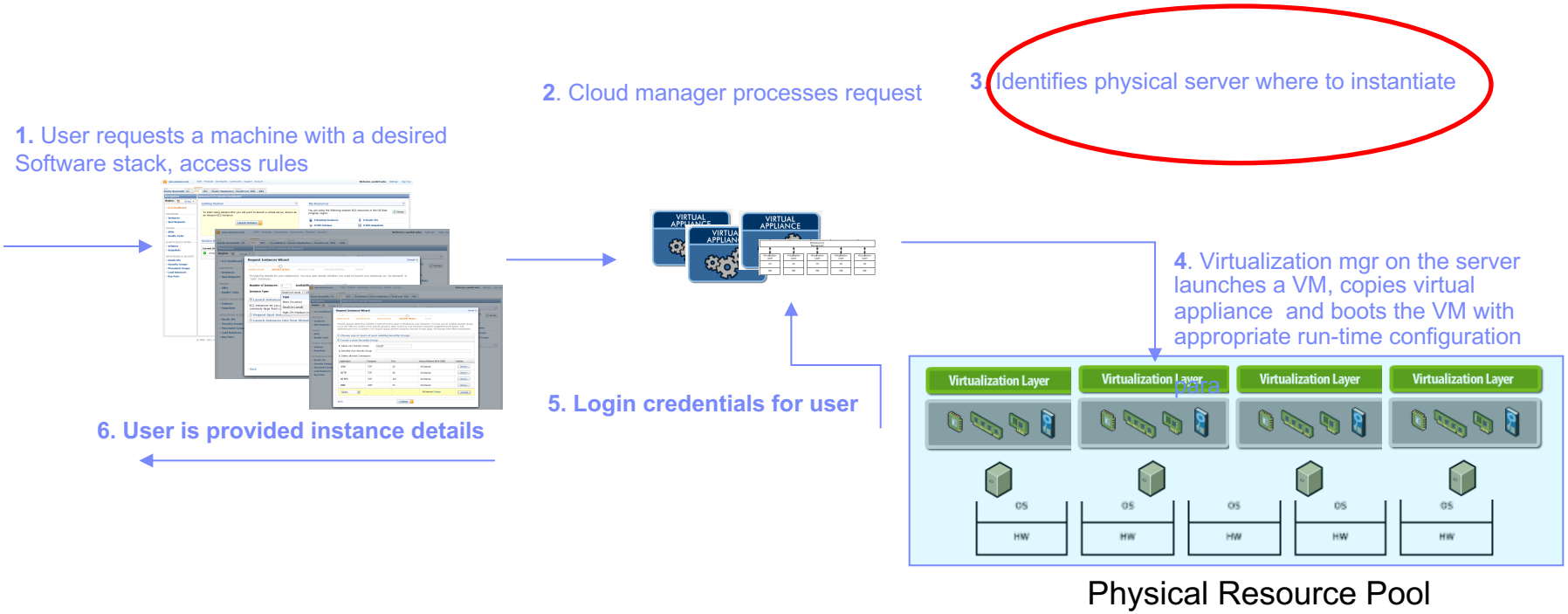


Figure 1: Migration timeline

Deconstructing Amazon EC2 request machine API

- User goes to Amazon EC2 portal and specifies desired parameters for a machine
  - Resource: CPU, mem, disk
  - Stack: OS and possibly with additional software
- Amazon AWS Cloud manager (resource pool manager) provisions the user request
  - Finds appropriate physical resource
  - Dispatches the request to virtualization manager on the identified resource
  - Cloud Manager invokes EC2 API to provisions the request
- Virtualization manager on physical server
  - Copies the pre-built software stack (virtual appliance)
  - Provisions a guest VM and configures parameters (IP address, access rules,…) at run/boot time
- Cloud manager returns login credentials to user



Key building blocks

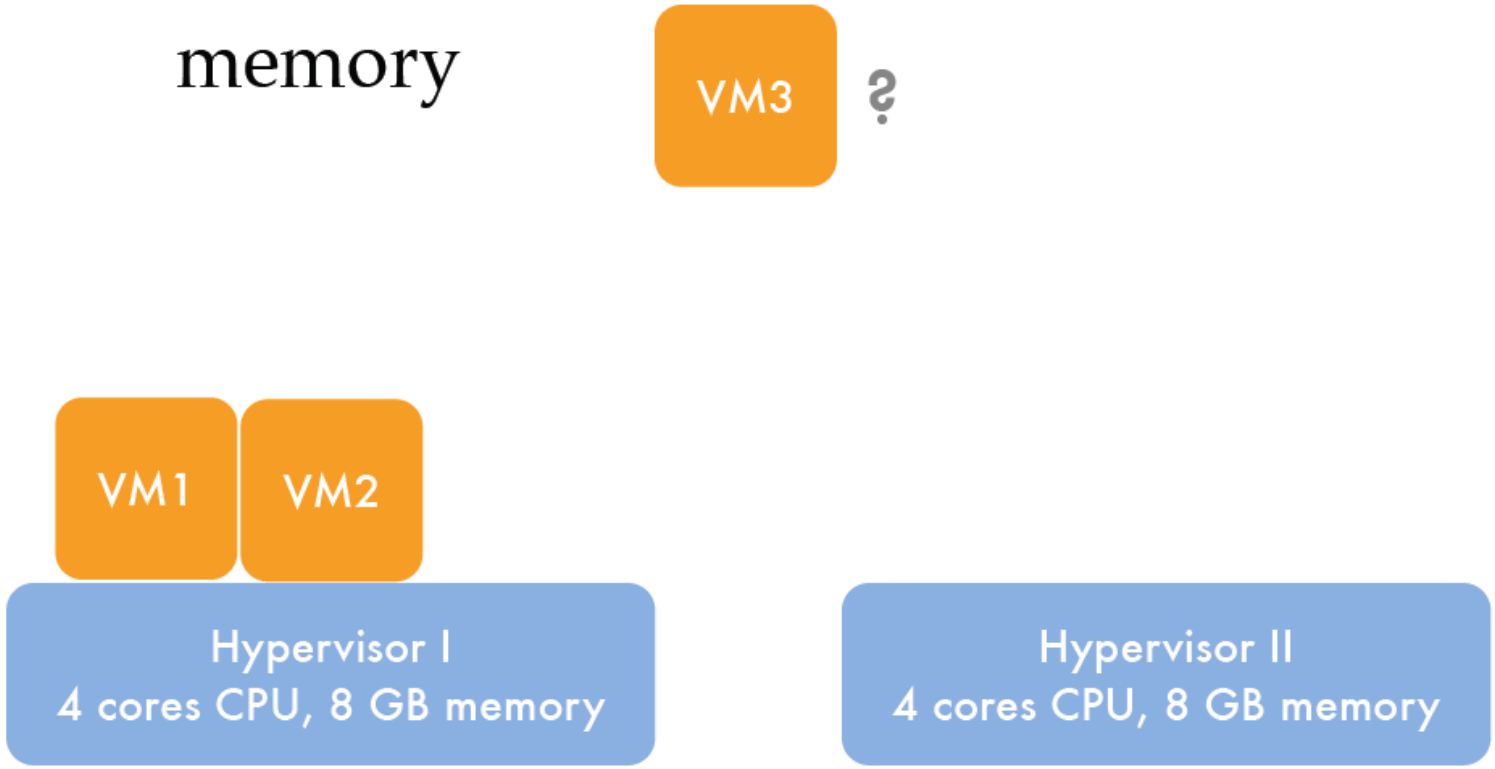
- Cloud manager
- Virtual machine
- Virtual appliance
- Configuring virtual appliance at run to meet the configuration parameters

VM Placement

- When a user requests a VM, which hypervisor should the cloud provider choose to place/run the VM?
- Example 1: Small instance 1 core CPU, 2 GB memory. How many small VM instances can this cloud support?

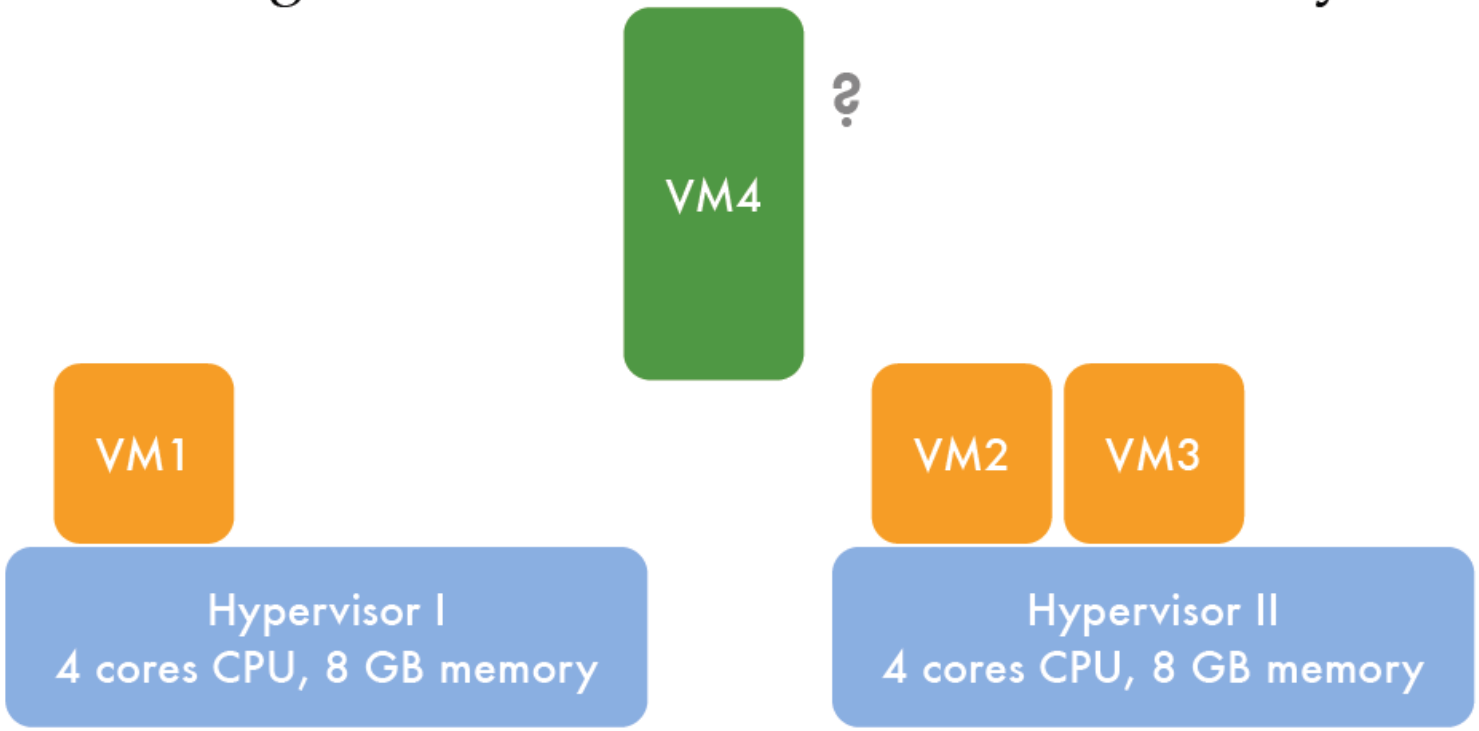
New request comes in

- Small instance 1 core CPU, 2 GB memory



New large instance comes in?

- Small instance 1 core CPU, 2 GB memory
- Large instance 2 core CPU, 4 GB memory

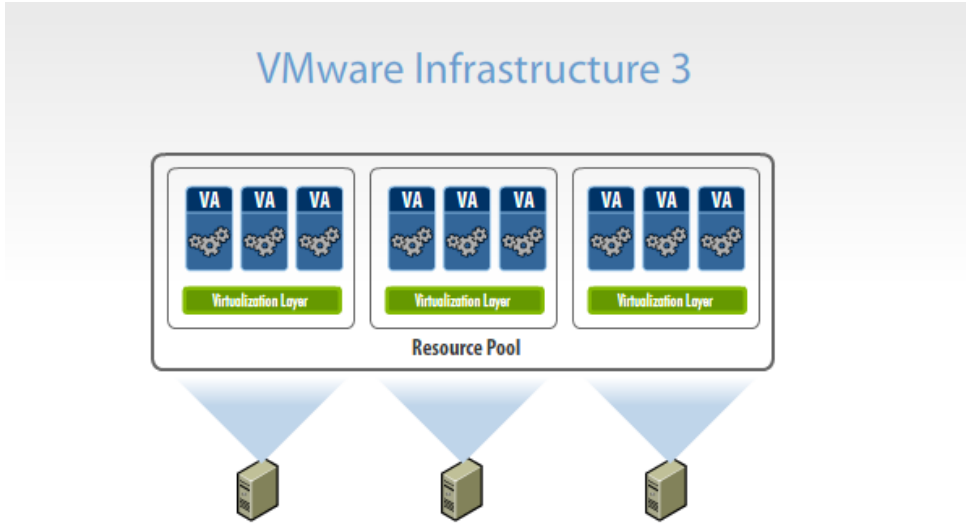


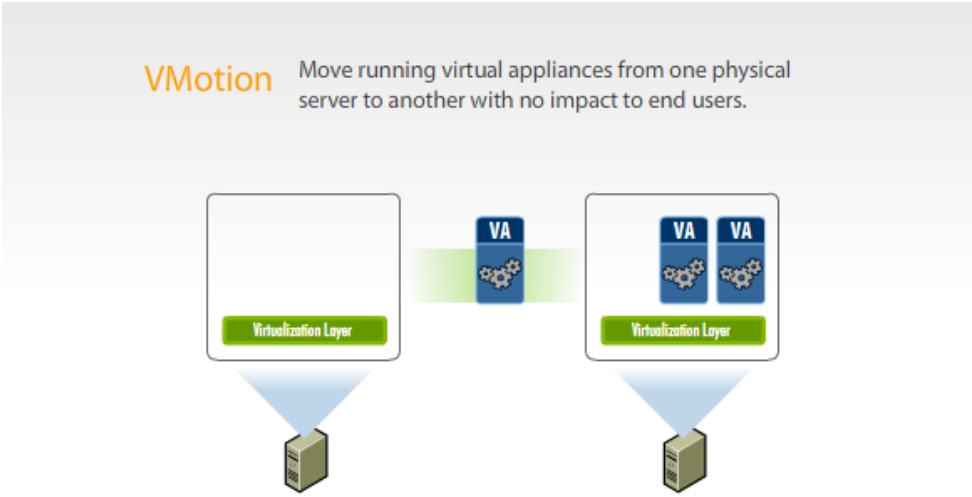
What to consider?

- Pack one hypervisor up at a time?
- Pack each hypervisor with only one instance type?
  - Fragmentation?
- What other resources do you need to consider?
  - Disk space
  - Energy consumption
  - Workload locality
  - High availability

Another important application of Virtualization

- Note that virtual machines are created on demand by issuing requests to hypervisor
- Virtual machines can be moved from one physical server to another in real time!
- VMWare vMotion management software lets one move running virtual machines in real time from one server to another
  - Opens up lot of interesting scenarios
  - Zero down-time maintenance and/or upgrades
  - Dynamic workload balancing

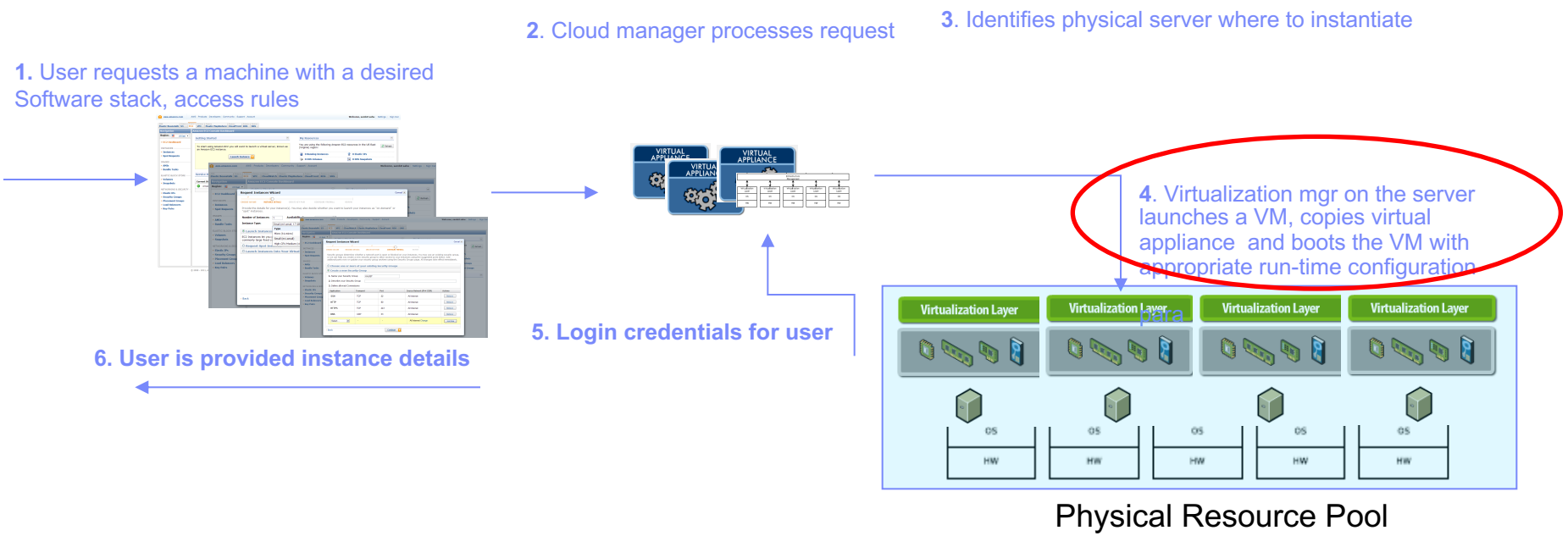




Moving all VMs from server A to server B

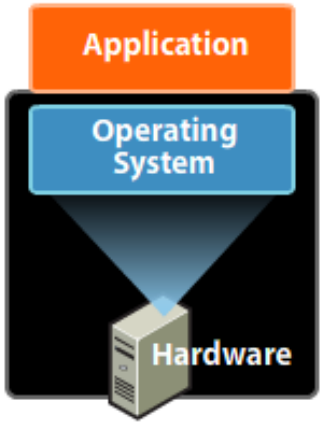
Deconstructing Amazon EC2 request machine API

- User goes to Amazon EC2 portal and specifies desired parameters for a machine
  - Resource: CPU, mem, disk
  - Stack: OS and possibly with additional software
- Amazon AWS Cloud manager (resource pool manager) provisions the user request
  - Finds appropriate physical resource
  - Dispatches the request to virtualization manager on the identified resource
  - Cloud Manager invokes EC2 API to provisions the request
- Virtualization manager on physical server
  - Copies the pre-built software stack (virtual appliance)
  - Provisions a guest VM and configures parameters (IP address, access rules,…) at run/boot time
- Cloud manager returns login credentials to user

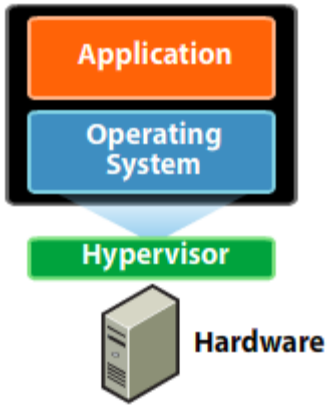


Virtual Appliance

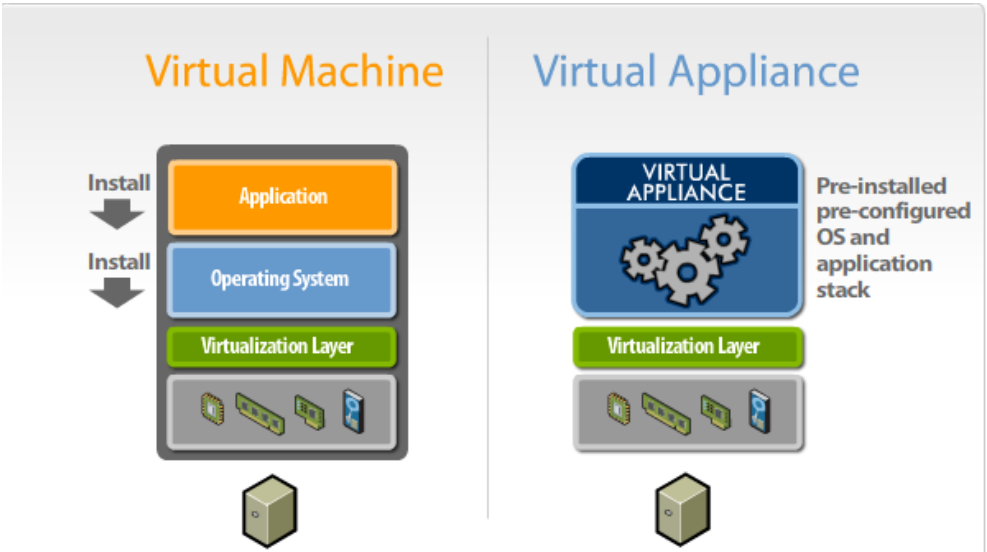
- How was Amazon EC2 able to start a (virtual) server with a software stack such as Operating System (win203, SUSE linux 32-bit, LAMP stack etc.) almost instantly?
  - User is able to choose from a list of available pre-built stack
- Virtual Appliance
  - A virtual machine image file consisting of pre-built/installed software bundled with operating system
  - Built in a such a manner that virtual machine boots from this pre-bundled stack
  - Install once and replicate many times
- Benefit
  - No need to install the software as long as same virtual machine technology used
  - Removes the need for time consuming installation and configuration of software



Traditional application software setup



allows the application and the operating system



VMWare Virtual Appliance Demo

- http://download3.vmware.com/media/vam/vam_demo.html

VMWare Virtual Appliance Marketplace

Images and virtual appliances

Image

- **Durable object** - not affected by VM execution
- Not bound to a runtime environment or hypervisor
- Suitable for sharing
- Consumes storage resources

Instance

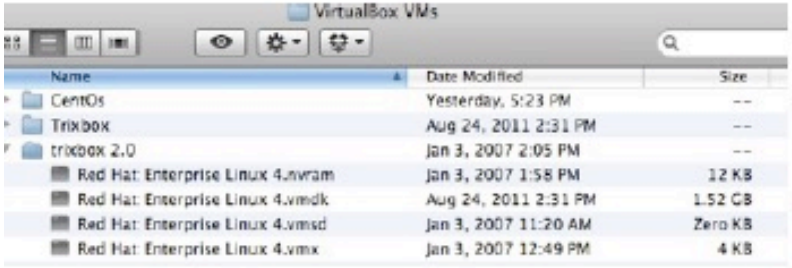
- **Volatile object** - modified as a side-effect of VM execution
- Bound to a specific runtime environment (e.g. IP addr)
- Consumes storage resources

VM

- **Temporal object** - logical computer running on a hypervisor platform
- Uses an instance disk as the backing store for its state
- Consumes server & network resources

What is an image?

- Provides a pre-built software stack for a virtual machine
- At VM creation time, it boots from this stack
- A typical VMWare based image consists of the following files



Extension	File name	Description
.vmdk	<vmname>.vmdk	Stores the contents of the virtual machine's hard disk drive.
.nvram	<vmname>.nvram	Stores the state of the virtual machine's BIOS.
.vmx	<vmname>.vmx	Stores the setting of the virtual machine.
.vmxf	<vmname>.vmxf	Stores supplemental configuration information
.vswp	<vmname>.vswp	Swap file

Managing instances

- Start
- Stop
- Suspend
- Resume
- Migrate
- Snapshot
- Clone

Image library optimizations

- Provisioning optimizations
  - How long did it take to provision an image?
- Storage optimizations
  - 90% of images are the same
  - Exploit redundancy among large numbers of images
- Active research areas

Managing instances

- Start
- Stop
- Snapshot
- Clone
- Suspend
- Resume
- Live Migrate

How does suspend-resume work?

- Suspend
  - Hypervisor writes the VM's memory state to disk in a file
  - In VMWare, the file is <vmname>.vmss
  - Then the VM is put into suspended state
- Resume
  - Hypervisor reads the image file
  - Hypervisor reads the memory state from file
  - Hypervisor runs the VM from where it last left off

Live Migration

- Let's watch first

- http://www.youtube.com/watch?v=dwPp35iyiw8

Live Migration

- VMWare (LAN Live Migration)
  - VMotion == Live Memory Migration
  - Storage VMotion == Live Storage Migration

Reference Materials

- ESXi: http://www.vmware.com/products/vsphere-hypervisor/
- vSphere: Mastering VMWare vSphere 5.5 – Scott Lowe
- vCloud: http://www.vmware.com/products/vcloud-suite

Monolith vs Micro-Services

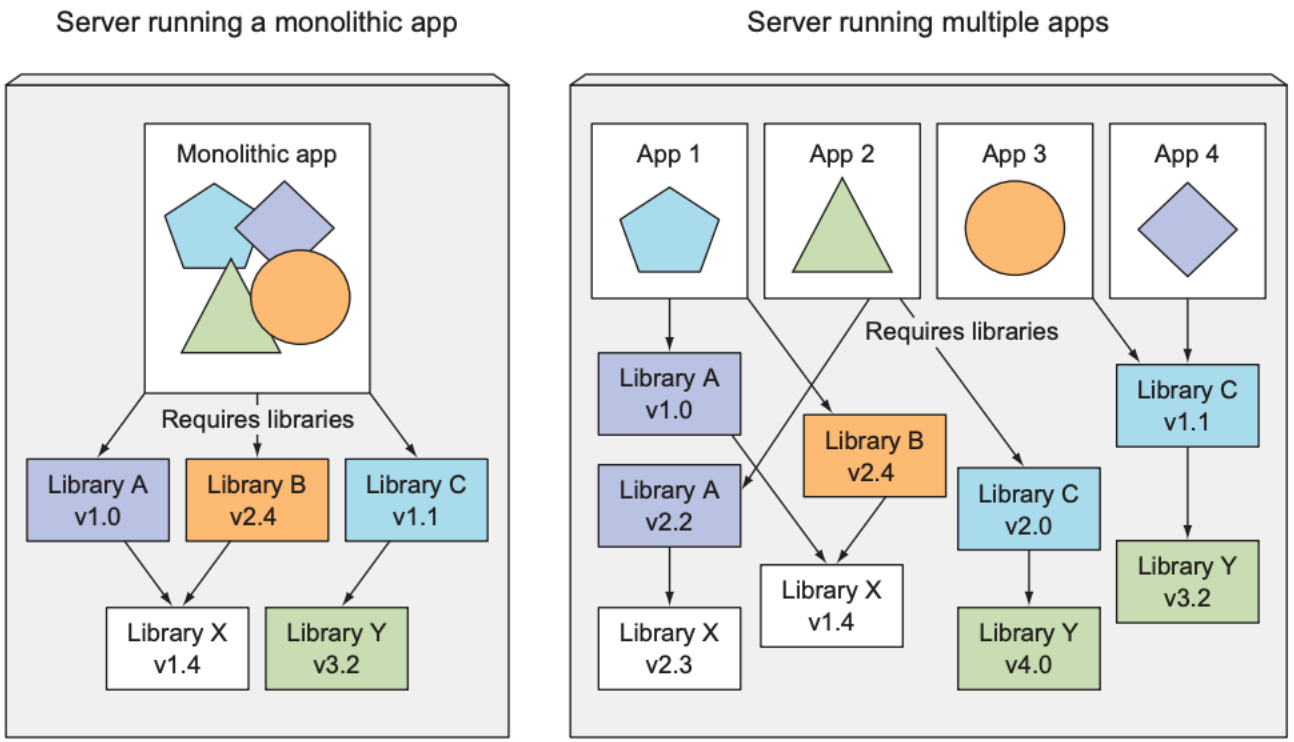


Figure 1.3 Multiple applications running on the same host may have conflicting dependencies.

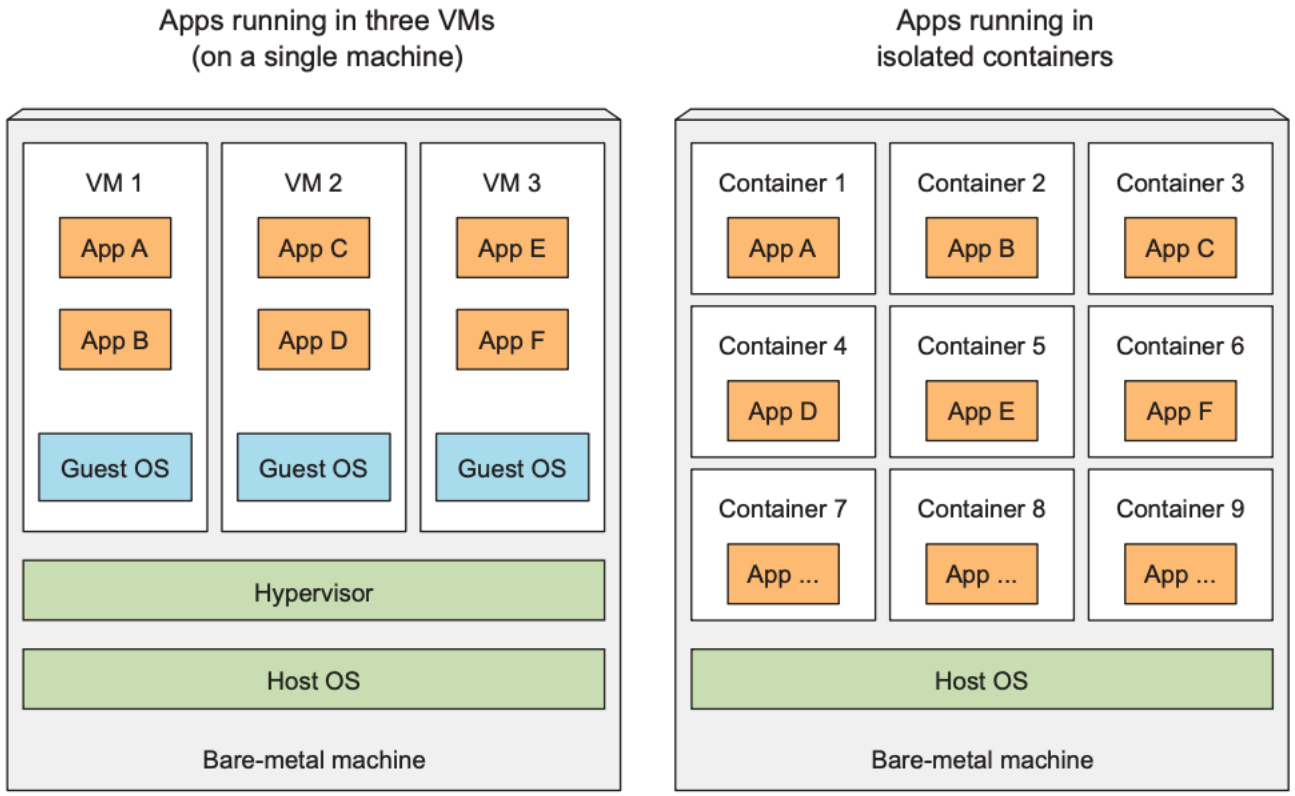


Figure 1.4 Using VMs to isolate groups of applications vs. isolating individual apps with containers

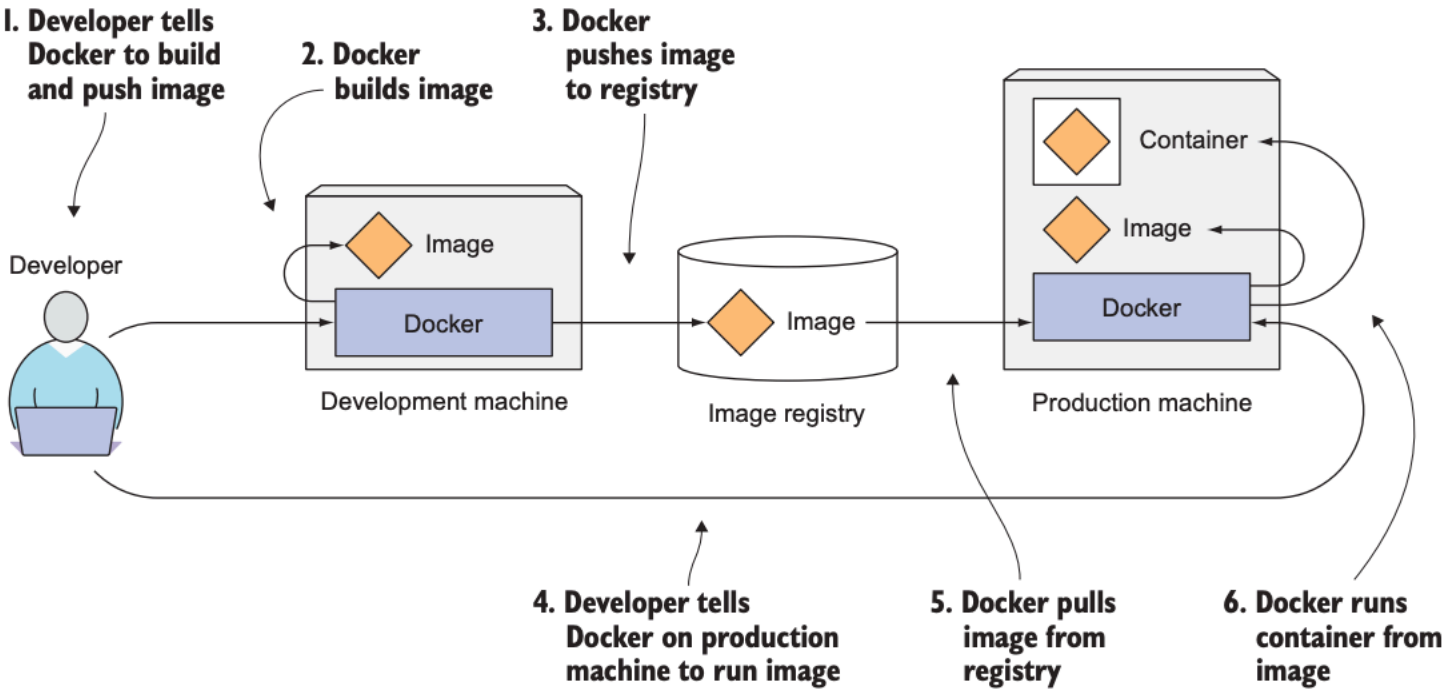
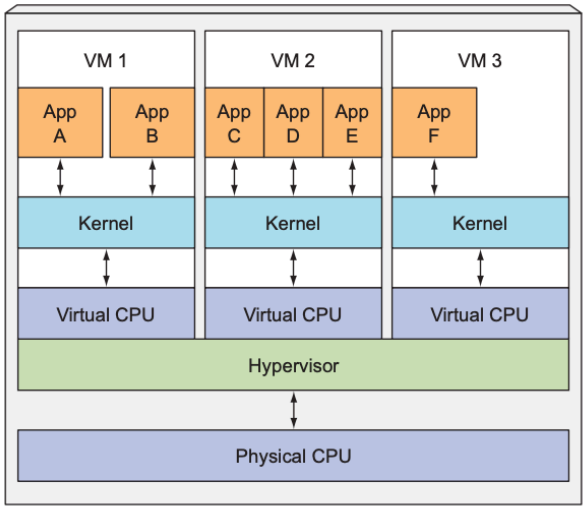
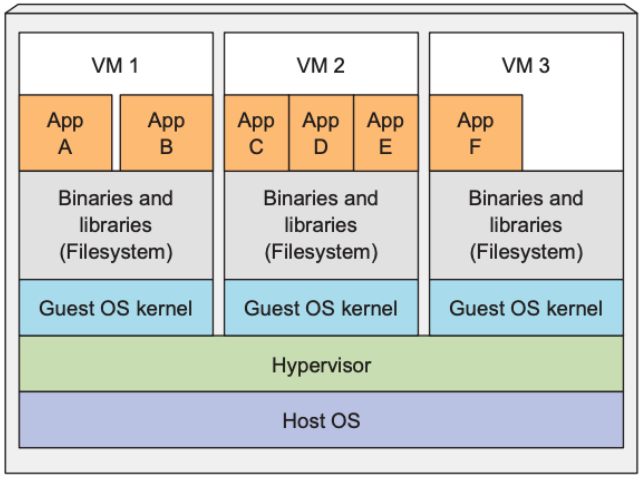


Figure 1.6 Docker images, registries, and containers

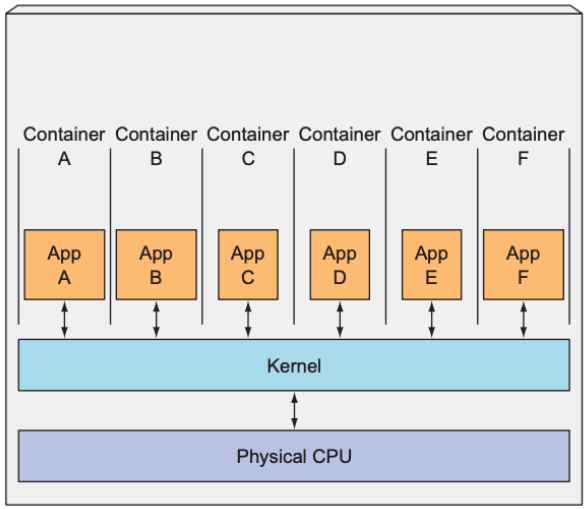
Apps running in multiple VMs



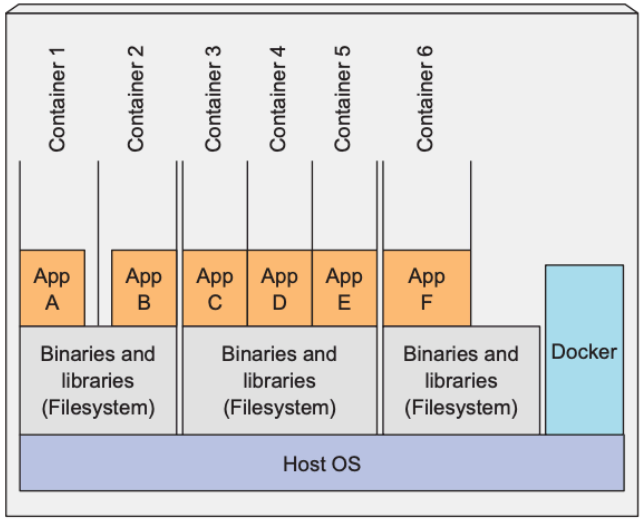
Host running multiple VMs



Apps running in isolated containers



Host running multiple Docker containers



Dockers and Containers

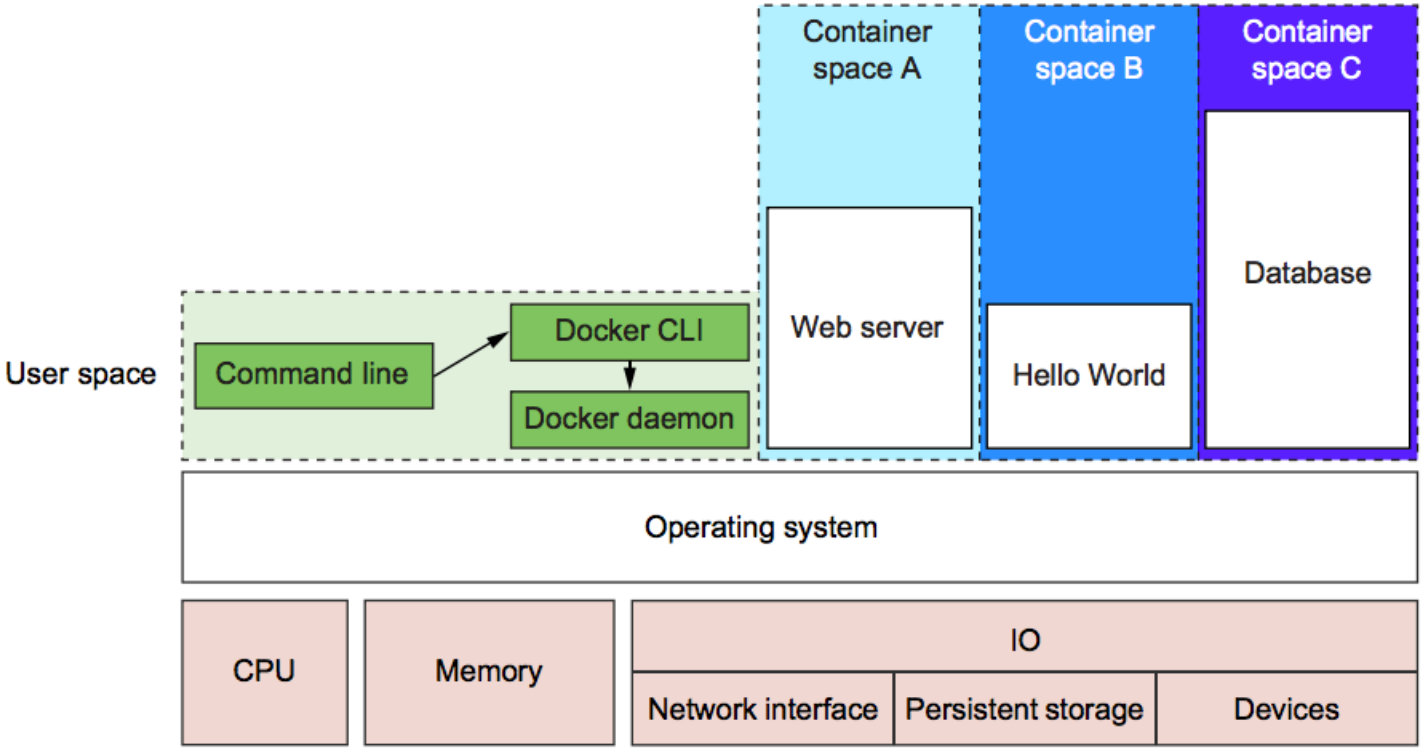


Figure 1.2 Docker running three containers on a basic Linux computer system

- More light weight and runs as user space processes
- Containers provide the separation