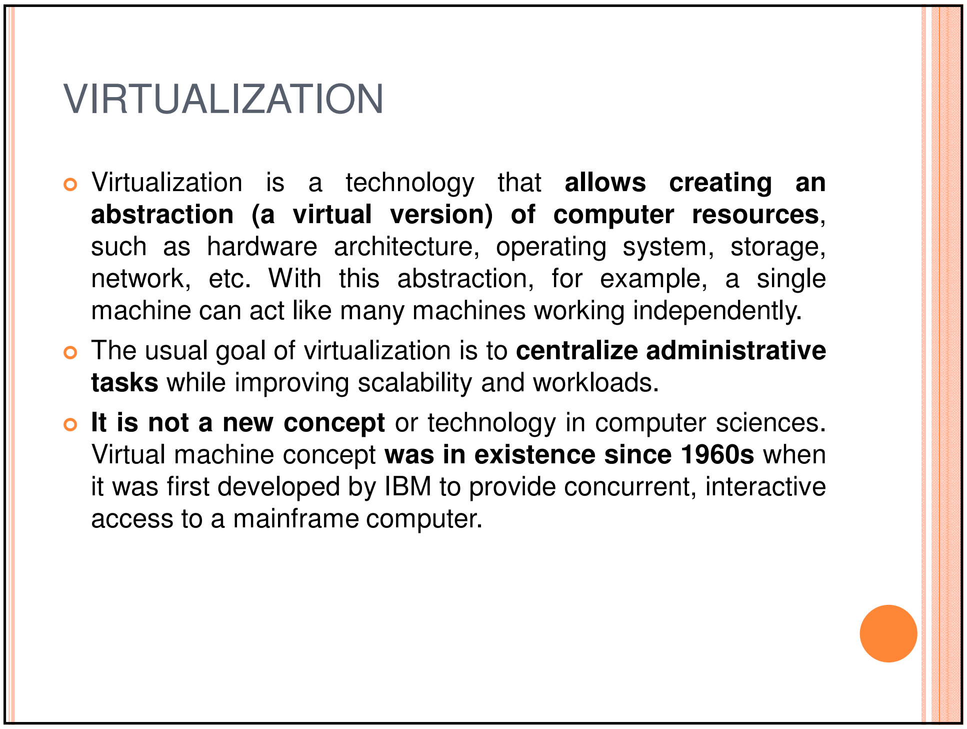
Understanding Abstraction and Virtualization

Víctor J. Sosa S.
vjsosa@tamps.cinvestav.mx



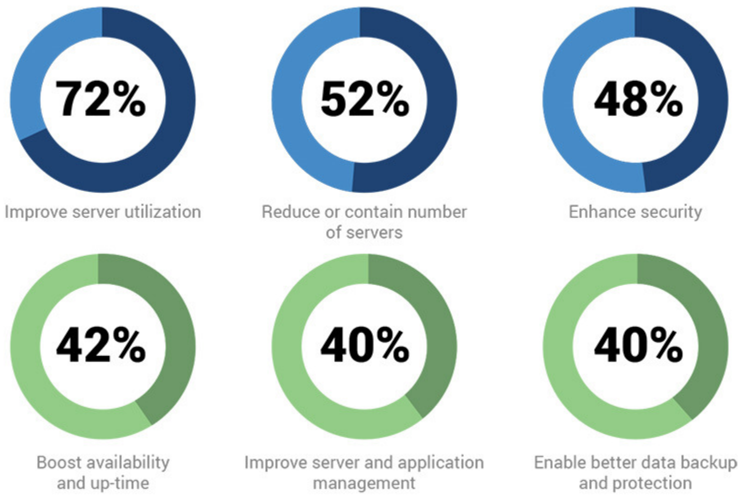
VIRTUALIZATION

- Virtualization is a technology that **allows creating an abstraction (a virtual version) of computer resources**, such as hardware architecture, operating system, storage, network, etc. With this abstraction, for example, a single machine can act like many machines working independently.
- The usual goal of virtualization is to **centralize administrative tasks** while improving scalability and workloads.
- **It is not a new concept** or technology in computer sciences. Virtual machine concept **was in existence since 1960s** when it was first developed by IBM to provide concurrent, interactive access to a mainframe computer.

VIRTUALIZATION

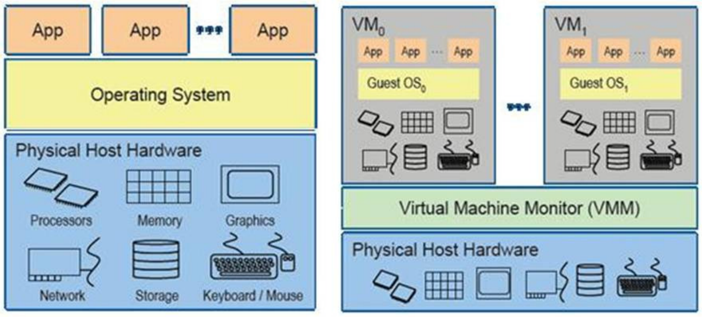
- **Reduce power consumption (**air conditioning needs, trim the building space and land requirements associated with server farm growth),
- **High availability for critical applications**, and streamlines application deployment and migrations,
- **Simplifies IT operations** and allow IT organizations to **respond faster to changing business demands**,
- **Greenhouse gas reduction** The socio-political ramifications of global warming requiring good corporate citizens **to meet** targets, creates an added incentive for virtualization.

Traditional Benefits of Virtualization



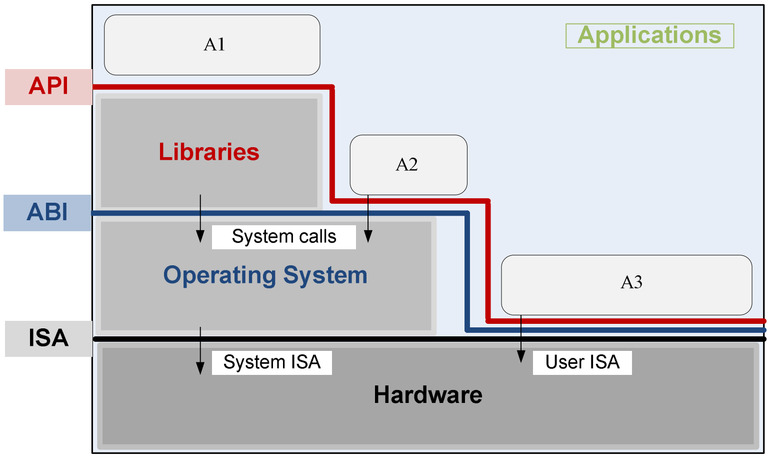
Source: http://www.vmware.com

Example of use of a Virtual Machine Monitor (VMM)



Without VMs: A Single OS owns all hardware resources

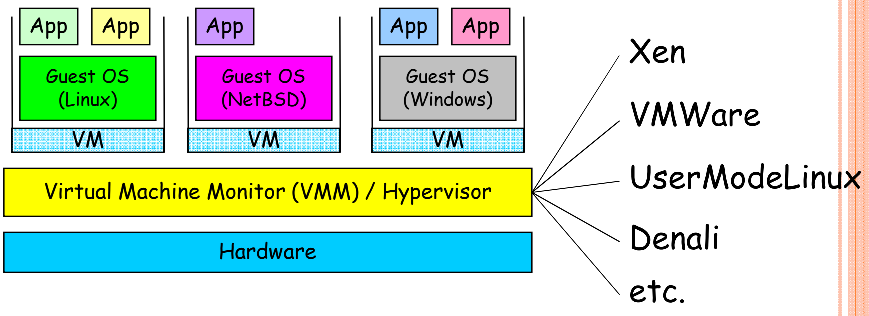
With VMs: Multiple OSs share hardware resources



Application Programming Interface, Application Binary Interface, and Instruction Set Architecture . An application uses library functions (A1), makes system calls (A2), and executes machine instructions (A3).

Virtual Machine Monitor: examples

- VM technology allows multiple virtual machines to run on a single physical machine.



App App App App App

Guest OS (Linux)

Guest OS (NetBSD)

Guest OS (Windows)

VM VM VM

Virtual Machine Monitor (VMM) / Hypervisor

Hardware

Xen

VMWare

UserModeLinux

Denali

etc.

Performance: Para-virtualization (e.g. Xen) is very close to raw physical performance!

Virtualization scenarios

a) **Server Consolidation**: To consolidate workloads of multiple under-utilized machines to fewer machines to save on hardware, management, and administration of the infrastructure.

b) **Application consolidation**: A legacy application might require newer hardware and/or operating systems. Fulfillment of the need of such legacy applications could be served well by virtualizing the newer hardware and providing its access to others.

c) **Sandboxing**: Virtual machines are useful to provide secure, isolated environments (sandboxes) for running foreign or less-trusted applications. Virtualization technology can, thus, help build secure computing platforms.

d) **Multiple execution environments**: Virtualization can be used to create mutiple execution environments (in all possible ways) and can increase the QoS by guaranteeing specified amount of resources.

VIRTUALIZATION SCENARIOS (CONT..)

d) **Virtual hardware**: It can provide the hardware one never had, e.g. Virtual SCSI drives, Virtual ethernet adapters, virtual ethernet switches and hubs, and so on.

e) **Multiple simultaneous OS**: It can provide the facility of having multiple simultaneous operating systems that can run many different kind of applications.

f) **Debugging**: It can help debug complicated software such as an operating system or a device driver by letting the user execute them on an emulated PC with full software controls.

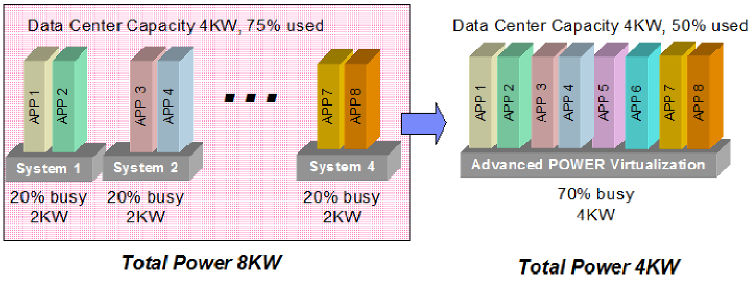
g) **Software Migration**: Eases the migration of software and thus helps mobility.

h) **Appliances**: Lets one package an application with the related operating environment as an appliance.

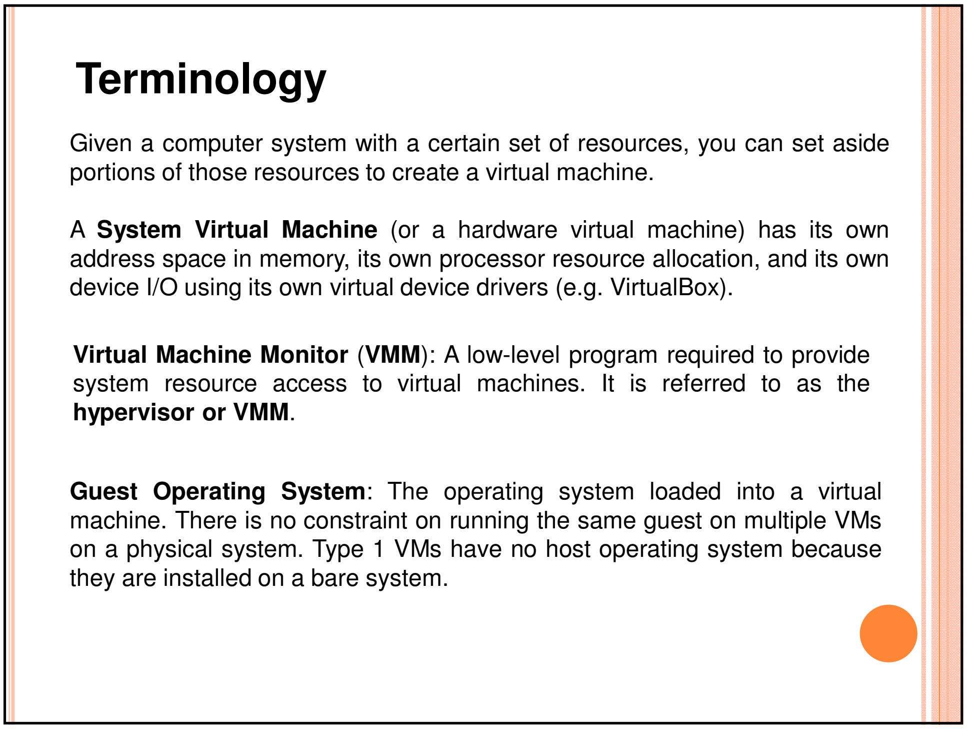
i) **Testing/QA***: Helps produce arbitrary test scenarios that are hard to produce in reality and thus eases the testing of software.

*QA: Quality Assurance

SERVER CONSOLIDATION



Server consolidation exploiting virtualization is a very effective tool in reducing energy costs



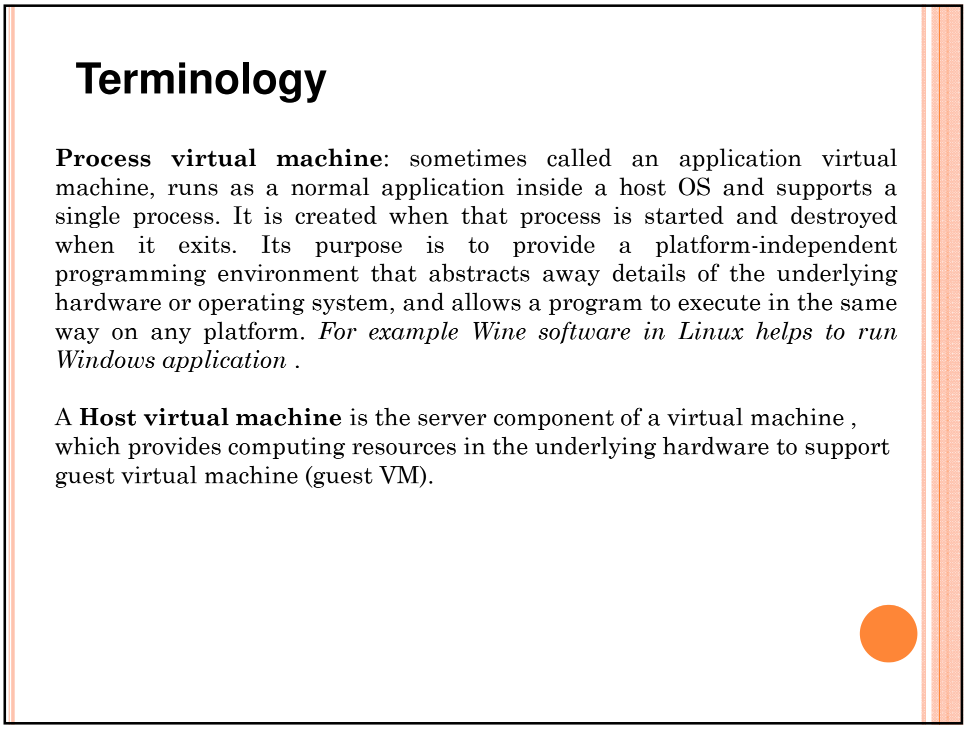
Terminology

Given a computer system with a certain set of resources, you can set aside portions of those resources to create a virtual machine.

A **System Virtual Machine** (or a hardware virtual machine) has its own address space in memory, its own processor resource allocation, and its own device I/O using its own virtual device drivers (e.g. VirtualBox).

Virtual Machine Monitor (**VMM**): A low-level program required to provide system resource access to virtual machines. It is referred to as the **hypervisor or VMM**.

Guest Operating System: The operating system loaded into a virtual machine. There is no constraint on running the same guest on multiple VMs on a physical system. Type 1 VMs have no host operating system because they are installed on a bare system.



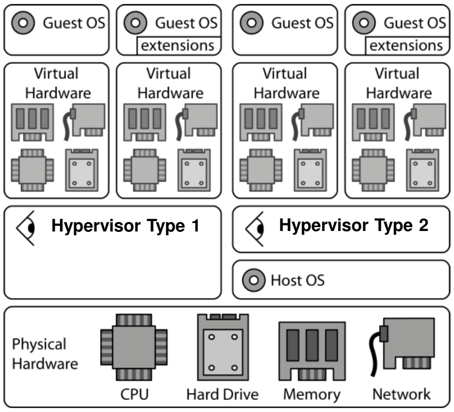
Terminology

Process virtual machine: sometimes called an application virtual machine, runs as a normal application inside a host OS and supports a single process. It is created when that process is started and destroyed when it exits. Its purpose is to provide a platform-independent programming environment that abstracts away details of the underlying hardware or operating system, and allows a program to execute in the same way on any platform. *For example Wine software in Linux helps to run Windows application .*

A **Host virtual machine** is the server component of a virtual machine , which provides computing resources in the underlying hardware to support guest virtual machine (guest VM).

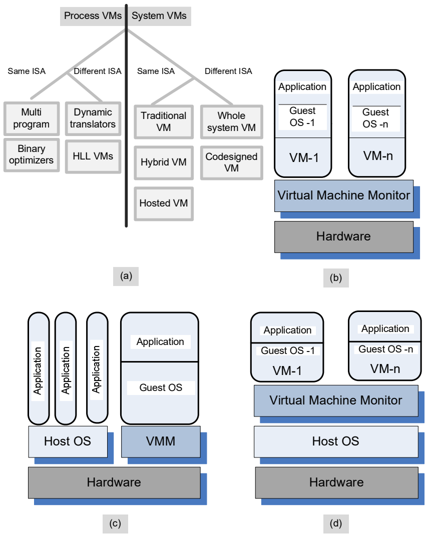
Hypervisor Main Approaches

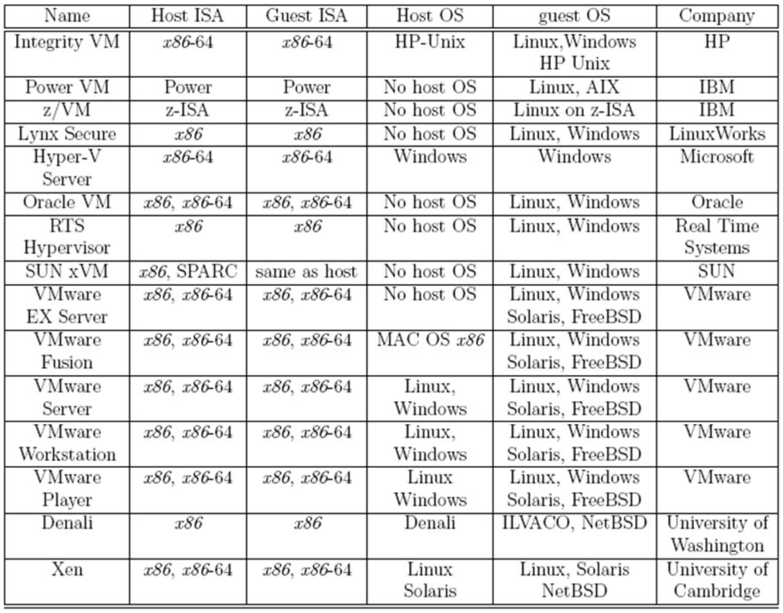
Examples:
Hyper-V
Xen
VMWare



Examples:
Oracle VirtualBox
VMWare Workstation
Virtual PC & Server
KVM

Traditional, Hybrid, and Hosted VMs

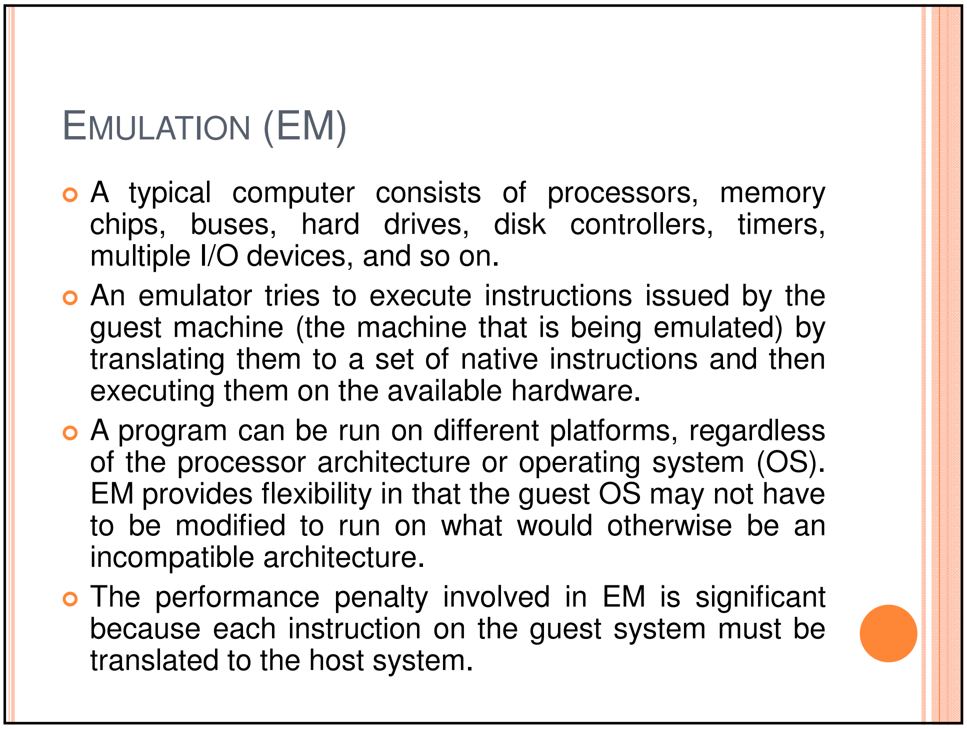




Name	Host ISA	Guest ISA	Host OS	guest OS	Company
Integrity VM	*x86-64*	*x86-64*	HP-Unix	Linux,Windows HP Unix	HP
Power VM	Power	Power	No host OS	Linux, AIX	IBM
z/VM	z-ISA	z-ISA	No host OS	Linux on z-ISA	IBM
Lynx Secure	*x86*	*x86*	No host OS	Linux, Windows	LinuxWorks
Hyper-V Server	*x86-64*	*x86-64*	Windows	Windows	Microsoft
Oracle VM	*x86*, *x86-64*	*x86*, *x86-64*	No host OS	Linux, Windows	Oracle
RTS Hypervisor	*x86*	*x86*	No host OS	Linux, Windows	Real Time Systems
SUN xVM	*x86*, SPARC	same as host	No host OS	Linux, Windows	SUN
VMware EX Server	*x86*, *x86-64*	*x86*, *x86-64*	No host OS	Linux, Windows Solaris, FreeBSD	VMware
VMware Fusion	*x86*, *x86-64*	*x86*, *x86-64*	MAC OS *x86*	Linux, Windows Solaris, FreeBSD	VMware
VMware Server	*x86*, *x86-64*	*x86*, *x86-64*	Linux, Windows	Linux, Windows Solaris, FreeBSD	VMware
VMware Workstation	*x86*, *x86-64*	*x86*, *x86-64*	Linux, Windows	Linux, Windows Solaris, FreeBSD	VMware
VMware Player	*x86*, *x86-64*	*x86*, *x86-64*	Linux Windows	Linux, Windows Solaris, FreeBSD	VMware
Denali	*x86*	*x86*	Denali	ILVACO, NetBSD	University of Washington
Xen	*x86*, *x86-64*	*x86*, *x86-64*	Linux Solaris	Linux, Solaris NetBSD	University of Cambridge

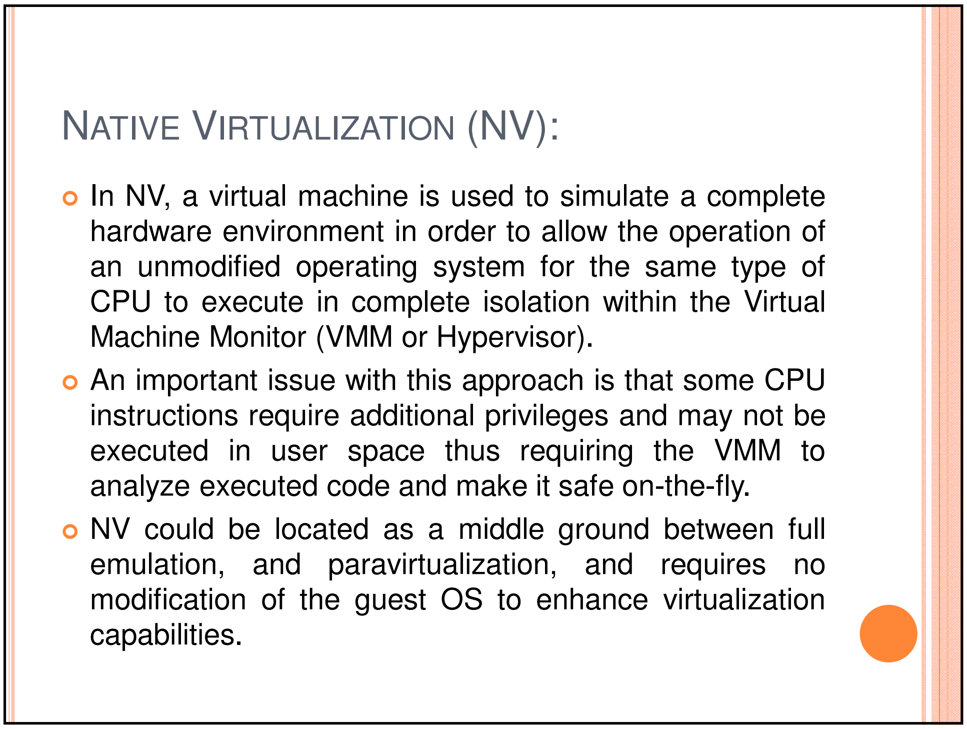
More virtualization techniques

- Virtualization techniques can be applied at different layers in a computer stack: hardware layer (including resources such as the computer architecture, storage, network, etc.), the operating system layer and application layer. Examples of **virtualization types** are:
  - **Emulation (EM)**
  - **Native Virtualization (NV)**
  - **Paravirtualization (PV)**
  - **Operating System Level Virtualization (OSLV)**
  - **Resource Virtualization (RV)**
  - **Application Virtualization (AV)**



EMULATION (EM)

- A typical computer consists of processors, memory chips, buses, hard drives, disk controllers, timers, multiple I/O devices, and so on.
- An emulator tries to execute instructions issued by the guest machine (the machine that is being emulated) by translating them to a set of native instructions and then executing them on the available hardware.
- A program can be run on different platforms, regardless of the processor architecture or operating system (OS). EM provides flexibility in that the guest OS may not have to be modified to run on what would otherwise be an incompatible architecture.
- The performance penalty involved in EM is significant because each instruction on the guest system must be translated to the host system.



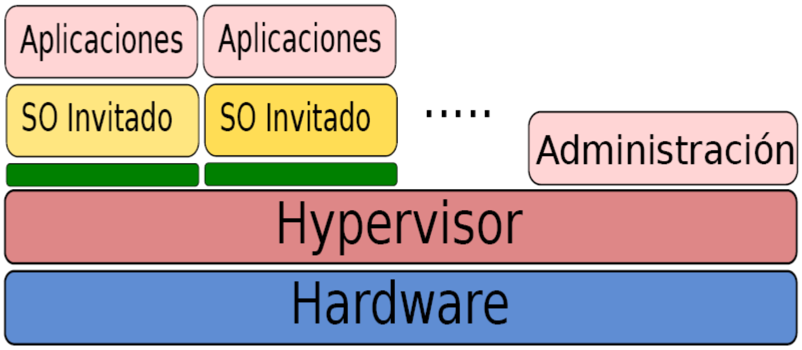
NATIVE VIRTUALIZATION (NV):

- In NV, a virtual machine is used to simulate a complete hardware environment in order to allow the operation of an unmodified operating system for the same type of CPU to execute in complete isolation within the Virtual Machine Monitor (VMM or Hypervisor).
- An important issue with this approach is that some CPU instructions require additional privileges and may not be executed in user space thus requiring the VMM to analyze executed code and make it safe on-the-fly.
- NV could be located as a middle ground between full emulation, and paravirtualization, and requires no modification of the guest OS to enhance virtualization capabilities.

Paravirtualization (PV)

- In this technique a modified guest OS is able to speak directly to the VMM.
- A successful paravirtualized platform may allow the VMM to be simpler (by relocating execution of critical tasks from the virtual domain to the host domain), and/or reduce the overall performance degradation of machine-execution inside the virtual-guest.
- Paravirtualization requires the guest operating system to be explicitly ported for the paravirtualization-API.
- A conventional OS distribution which is not paravirtualization-aware cannot be run on top of a paravirtualizing VMM.

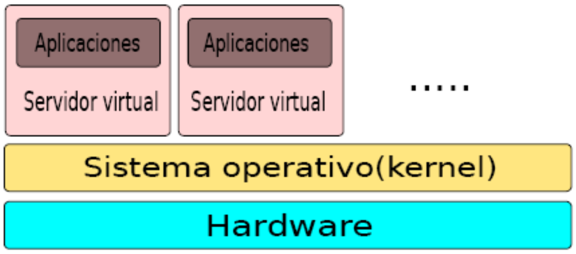
Paravirtualización

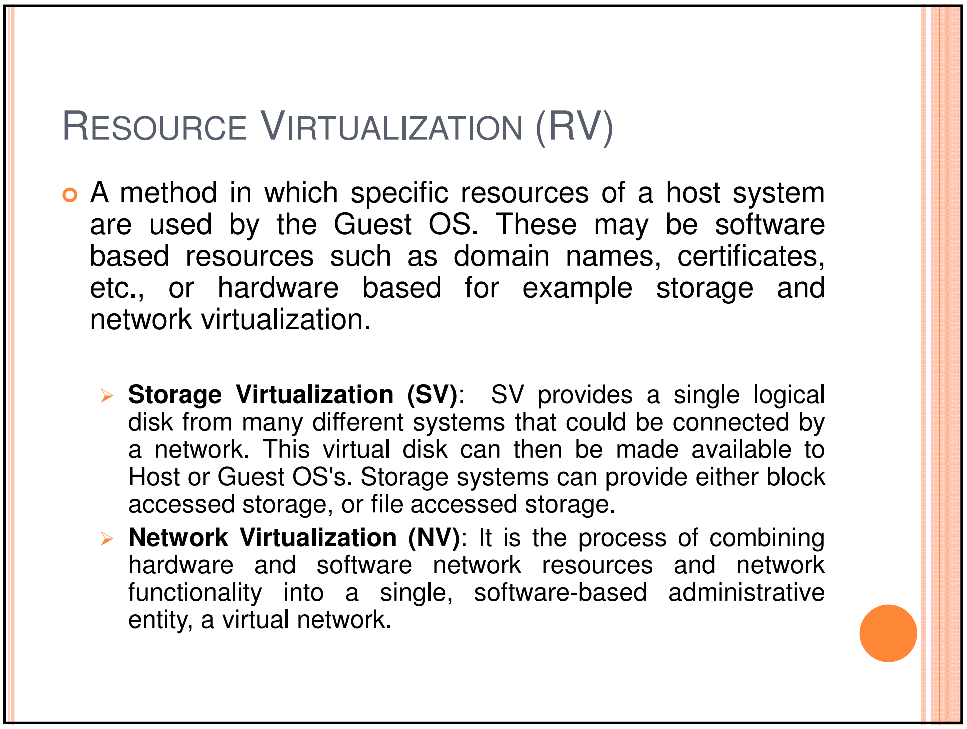


Operating System Level Virtualization (OSLV)

- A server virtualization method where the kernel of an operating system allows for multiple isolated user-space instances, instead of just one.
- It does provide the ability for user-space applications (that would be able to run normally on the host OS) to run in isolation from other software.
- Most implementations of this method can define resource management for the isolated instances.

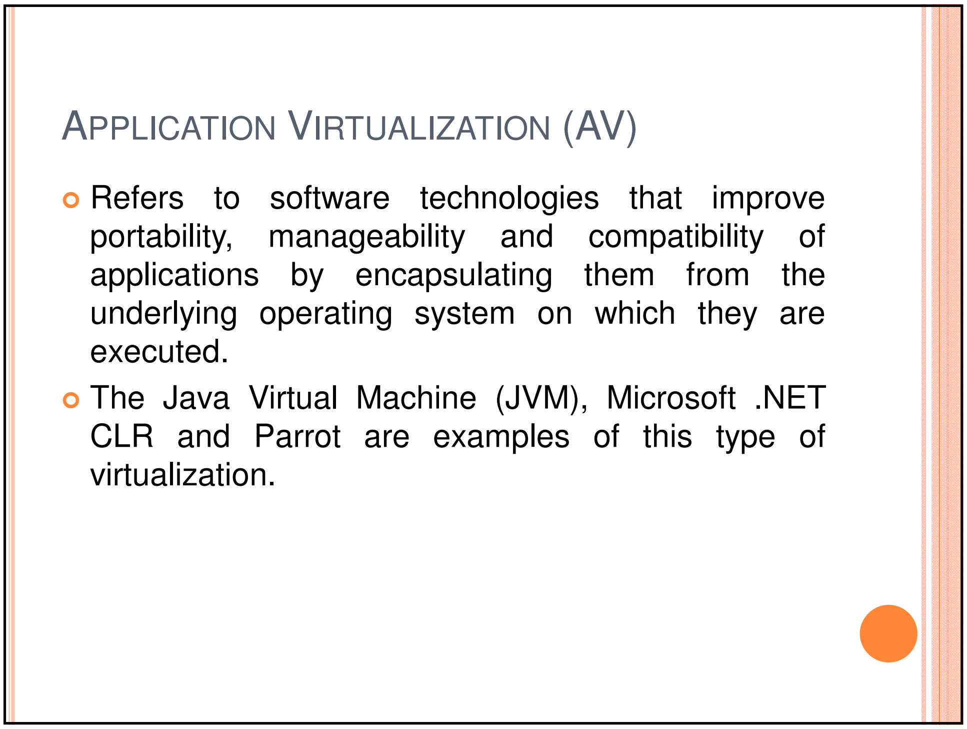
Virtualización a Nivel de Sistema Operativo





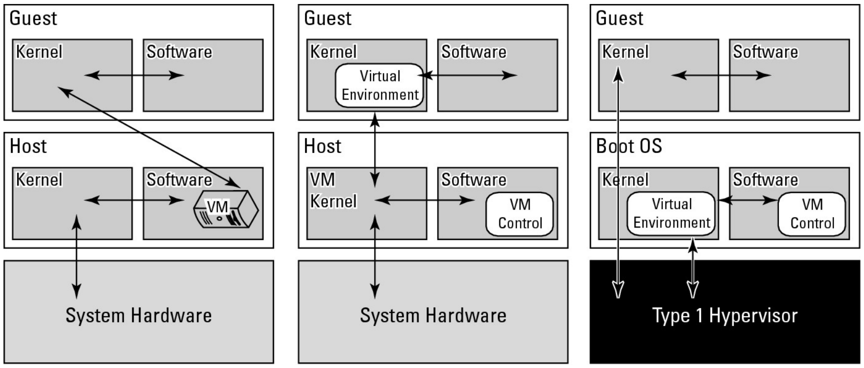
RESOURCE VIRTUALIZATION (RV)

- A method in which specific resources of a host system are used by the Guest OS. These may be software based resources such as domain names, certificates, etc., or hardware based for example storage and network virtualization.
  - **Storage Virtualization (SV)**: SV provides a single logical disk from many different systems that could be connected by a network. This virtual disk can then be made available to Host or Guest OS's. Storage systems can provide either block accessed storage, or file accessed storage.
  - **Network Virtualization (NV)**: It is the process of combining hardware and software network resources and network functionality into a single, software-based administrative entity, a virtual network.



APPLICATION VIRTUALIZATION (AV)

- Refers to software technologies that improve portability, manageability and compatibility of applications by encapsulating them from the underlying operating system on which they are executed.
- The Java Virtual Machine (JVM), Microsoft .NET CLR and Parrot are examples of this type of virtualization.

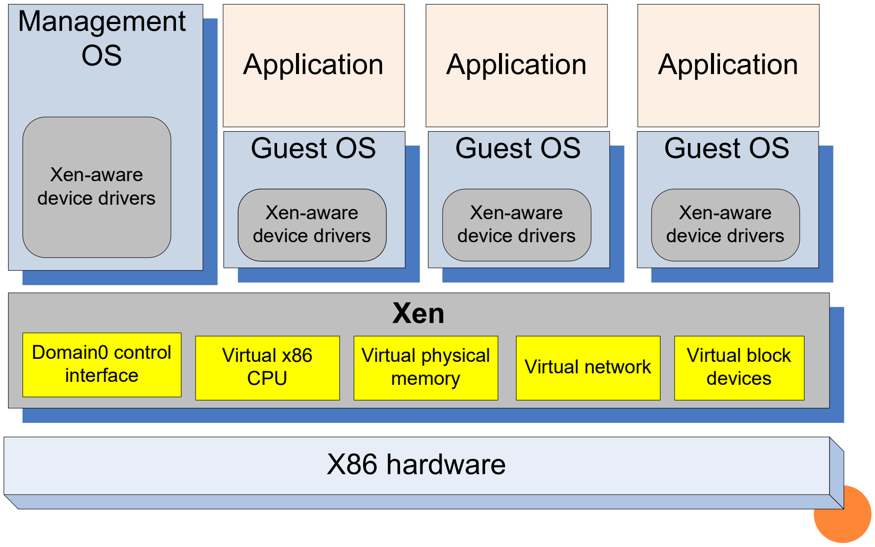

Guest
Kernel
Software
Host
Kernel
Software
VM
System Hardware

Guest
Kernel
Virtual Environment
Software
Host
VM Kernel
Software
VM Control
System Hardware

Guest
Kernel
Software
Boot OS
Kernel
Virtual Environment
Software
VM Control
Type 1 Hypervisor

Emulation **Paravirtualization** **Full Virtualization**

XEN


Management OS
Xen-aware device drivers
Application
Guest OS
Xen-aware device drivers
Application
Guest OS
Xen-aware device drivers
Application
Guest OS
Xen-aware device drivers
Xen
Domain0 control interface
Virtual x86 CPU
Virtual physical memory
Virtual network
Virtual block devices
X86 hardware

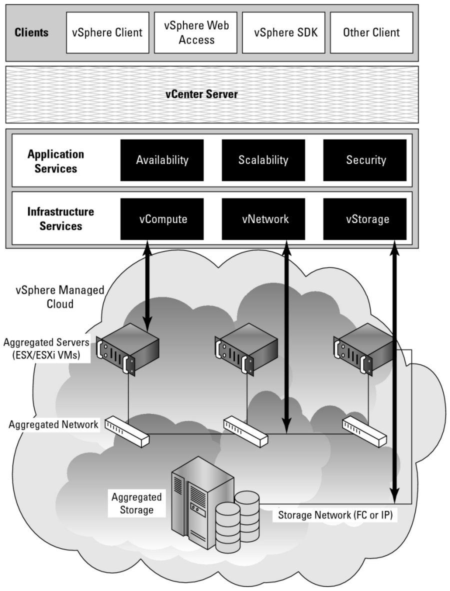
Example: VMware vSphere

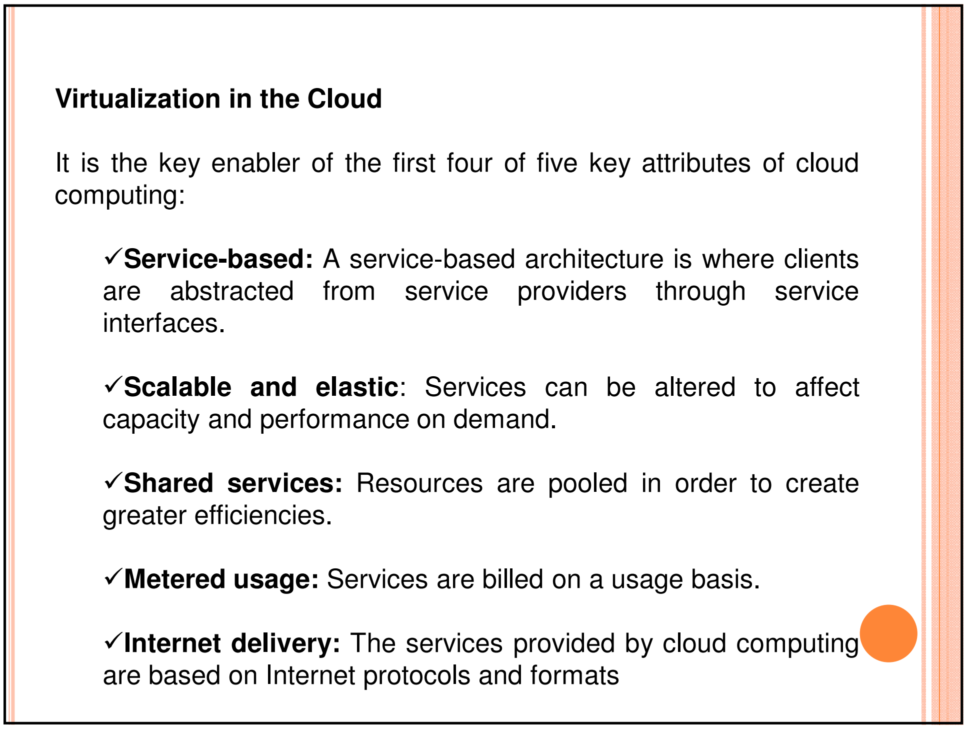
A management infrastructure framework that virtualizes system, storage, and networking hardware to create cloud computing infrastructures.

vSphere is the branding for a set of management tools and a set of products previously labeled **VMWare Infrastructure**.

it provides a set of services that applications can use to access cloud resources, including these:

- **VMware vCompute:** A service that aggregates servers into an assignable pool
- **VMware vStorage:** A service that aggregates storage resources into an assignable pool
- **VMware vNetwork:** A service that creates and manages virtual network interfaces
- **Application services:** Such as HA (High Availability) and Fault Tolerance
- **vCenter Server:** A provisioning, management, and monitoring console for Vmware cloud infrastructures





Virtualization in the Cloud

It is the key enabler of the first four of five key attributes of cloud computing:

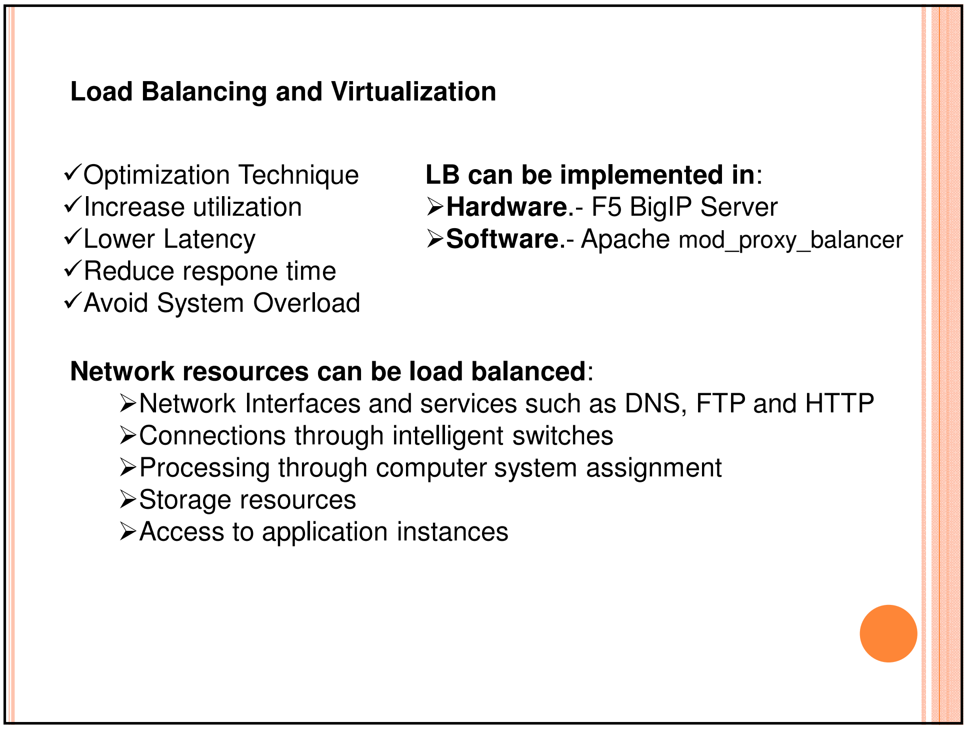
✓**Service-based:** A service-based architecture is where clients are abstracted from service providers through service interfaces.

✓**Scalable and elastic**: Services can be altered to affect capacity and performance on demand.

✓**Shared services:** Resources are pooled in order to create greater efficiencies.

✓**Metered usage:** Services are billed on a usage basis.

✓**Internet delivery:** The services provided by cloud computing are based on Internet protocols and formats



Load Balancing and Virtualization

✓Optimization Technique
✓Increase utilization
✓Lower Latency
✓Reduce respone time
✓Avoid System Overload

LB can be implemented in:
➢**Hardware**.- F5 BigIP Server
➢**Software**.- Apache mod_proxy_balancer

Network resources can be load balanced:

- ➢Network Interfaces and services such as DNS, FTP and HTTP
- ➢Connections through intelligent switches
- ➢Processing through computer system assignment
- ➢Storage resources
- ➢Access to application instances

Scheduling algorithms in use today are **round robin** and **weighted round robin**, fastest response time, least connections and weighted least connections, and custom assignments based on other factors.

Advanced load balancing

Sophisticated load balancers are workload managers. They **determine the current utilization of the resources in their pool, the response time, the work queue length, connection latency and capacity, and other factors in order to assign tasks to each resource**.

Among the features you find in load balancers are polling resources for their health, the ability to bring standby servers online (priority activation), workload weighting based on a resource's capacity (asymmetric loading), HTTP traffic compression, TCP offload and buffering, security and authentication, and packet shaping using content filtering and priority queuing.

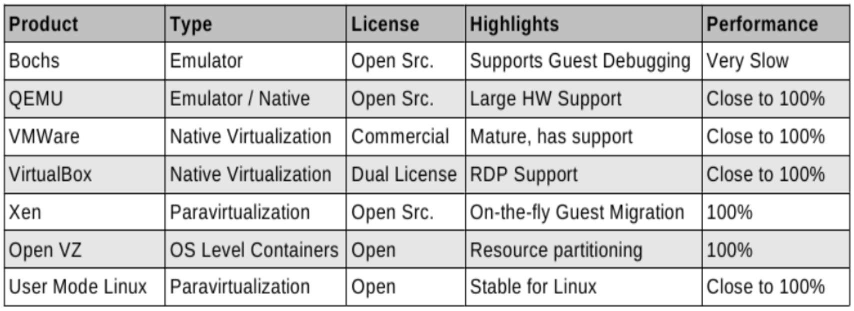
VIRTUALIZATION TOOLS

Type of Virtualization	Tools
Virtual Machines that give support: EM and/or NV	VMWare [113], VirtualBox [114], Microsoft Hyper-V Server [114], QEMU [115], Parallels [116], MS Virtual PC [117], Bochs [127]
Paravirtualization	Xen [118], User Mode Linux (UML) [119]
Virtualization at the OS level	FreeBSD Jails [120], OpenVZ [121], Virtuozzo [122], Linux-VServer [123], Solaris Zones (124)
Virtualization at the Programming Language Level (Applications).	Java Virtual Machine [125], Microsoft .NET CLR [126], Parrot [111]

SOME COMMERCIAL AND OPEN-SOURCE SOFTWARE (OSS) TOOLS FOR SERVER VIRTUALIZATION.

VMM	Type	Highlights	Guest performance	License
KVM	Full Virtualization	Assigns every VM as a regular Linux process	Close to native	Open Source
Xen	Paravirtualization	Supports VM migration on fly	Native	Open Source
VMware	Full Virtualization	Provides a mature product family to manage virtual infrastructure	Close to native	Commercial
Microsoft Hyper-V	Full Virtualization	Able to trap guest calls	Close to native	Commercial

SOME COMMERCIAL AND OPEN-SOURCE SOFTWARE (OSS) TOOLS FOR SERVER VIRTUALIZATION.



Product	Type	License	Highlights	Performance
Bochs	Emulator	Open Src.	Supports Guest Debugging	Very Slow
QEMU	Emulator / Native	Open Src.	Large HW Support	Close to 100%
VMWare	Native Virtualization	Commercial	Mature, has support	Close to 100%
VirtualBox	Native Virtualization	Dual License	RDP Support	Close to 100%
Xen	Paravirtualization	Open Src.	On-the-fly Guest Migration	100%
Open VZ	OS Level Containers	Open	Resource partitioning	100%
User Mode Linux	Paravirtualization	Open	Stable for Linux	Close to 100%