CAP470: Cloud Computing
Unit-3: Cloud Virtualisation

Dr. Manmohan Sharma

Professor

School of Computer Applications

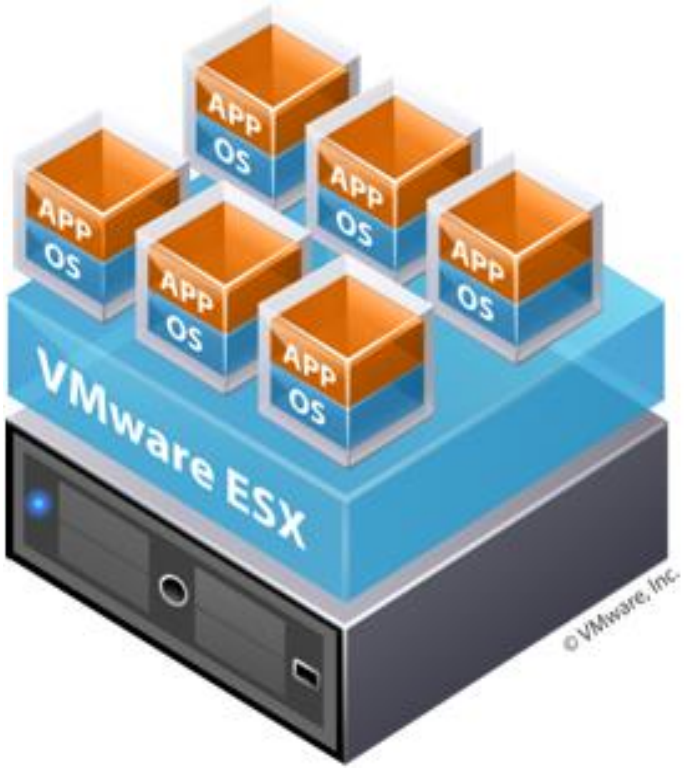
Lovely Professional University

Contents

- What is Virtualization?
- Need for Virtualization
- Types of Virtualization
- Pros and cons of virtualization
- System VM
- Process VM
- Virtual machine properties
- Virtual Machine monitor
- Interpretation and binary translation
- HLL VM - Hypervisors, Xen, KVM , VMWare, Virtual Box, Hyper-V

What is Virtualization?

- Virtualization is a Technology that transforms hardware into software.
- Virtualization allows to run multiple operating systems as virtual machines.
  - Each copy of an operating system is installed in to a virtual machine.



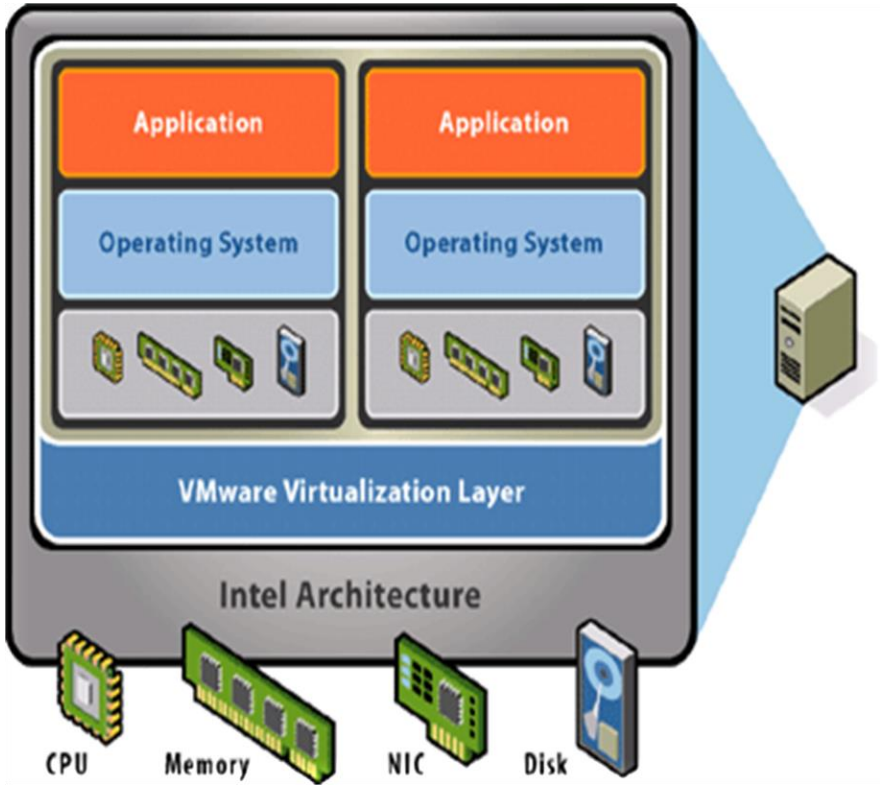
Need of Virtualization

- **Increased performance and computing capacity**
  - PCs are having immense computing power.
- **Underutilized hardware and software resources**
  - Limited use of increased performance & computing capacity.
- **Lack of space**
  - Continuous need for additional capacity.
- **Greening initiatives**
  - Reduce carbon footprints
  - Reducing the number of servers, reduce power consumption.
- **Rise of administrative costs**
  - Power and cooling costs are higher then IT equipment.

Types of Virtualization

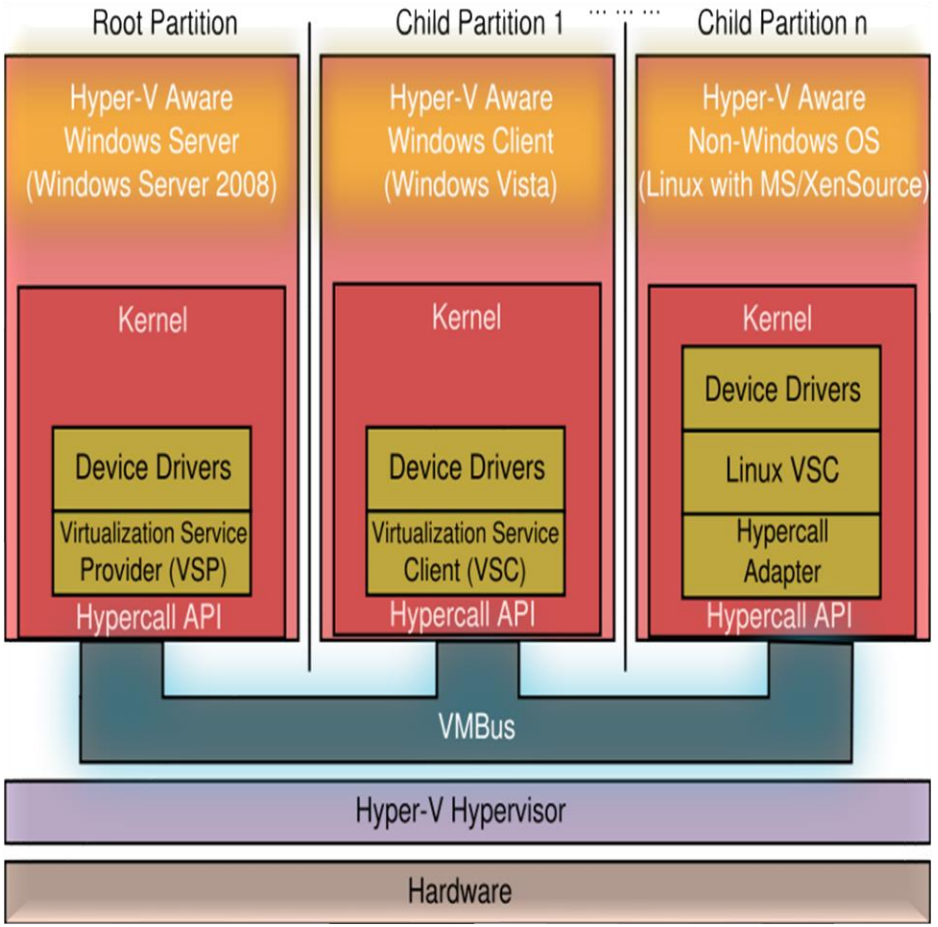
A. Full Virtualization

Full virtualization uses a special kind of software called a **hypervisor**. The hypervisor interacts directly with the physical server's CPU and disk space. It serves as a **platform** for the virtual servers OS.



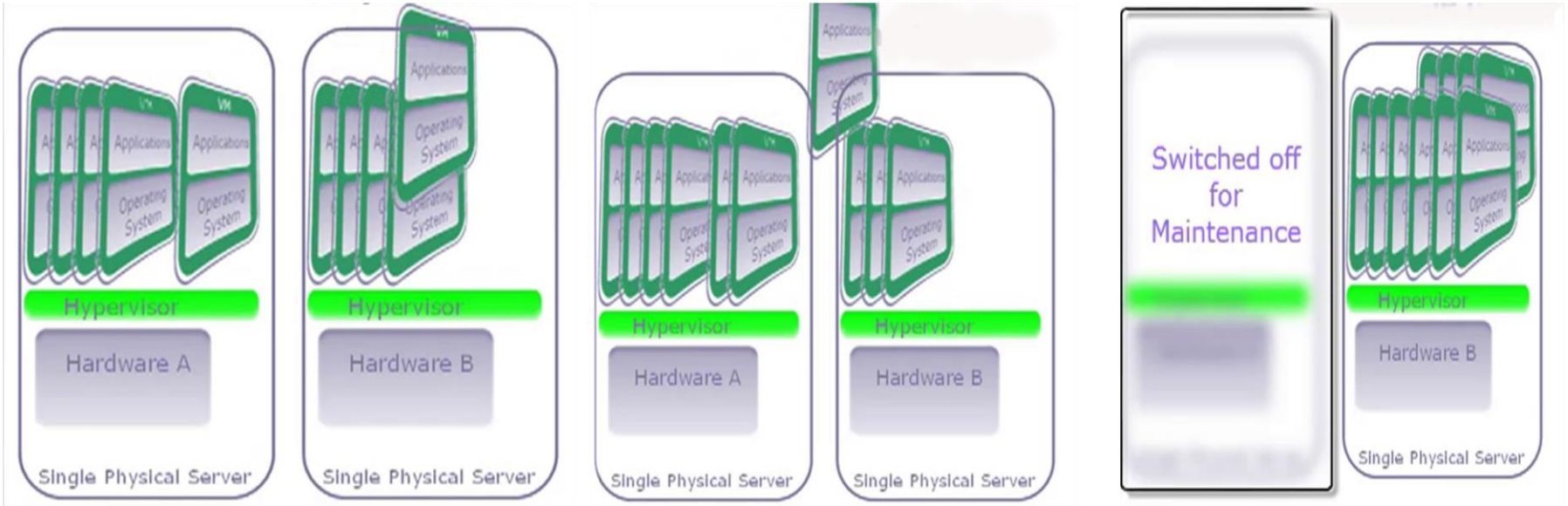
B. Para Virtualization

The para-virtualization approach is a little different than the full virtualization technique, the guest servers in a para - virtualization systems are aware of one another.



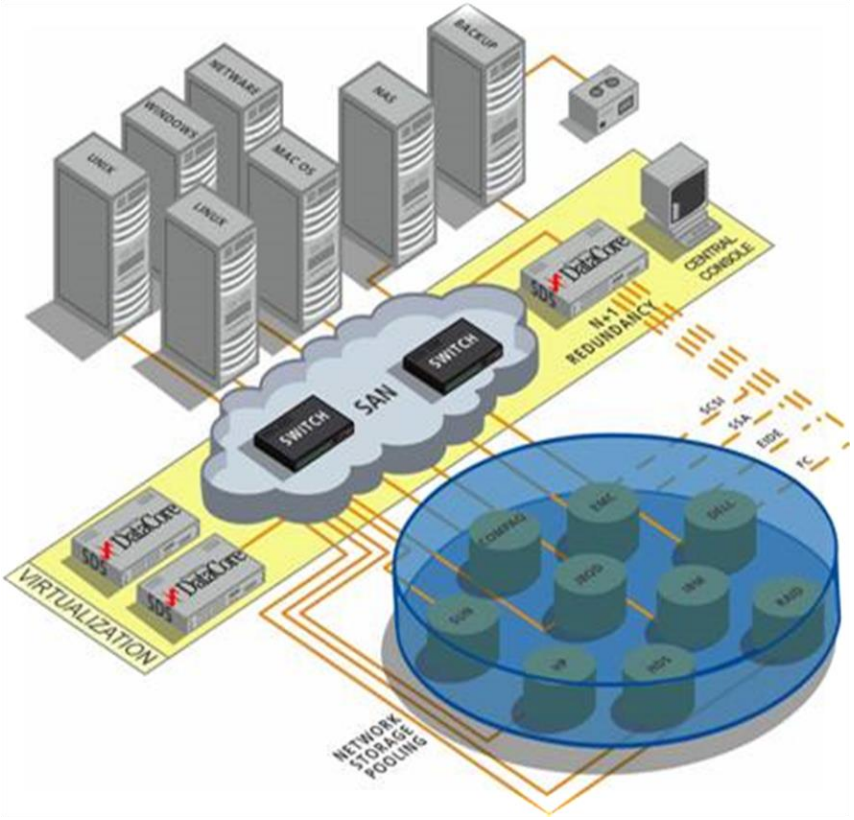
C. Server Virtualization

- The VMs created on one physical server can be run another physical server.
- Failure in one physical server will not effect the VMs installed on that.



D. Storage Virtualization

Storage Virtualization is the next frontier in Storage Advances that **aims to provide a layer of** abstraction to reduce complexity



E. Software Virtualization

Software virtualization is the virtualization of applications or computer programs.

Advantages of Virtualization

1. Efficient resource utilization

- Low resource utilization of each server machine
- Low total cost of ownership (TCO)
  - Low cost of infrastructure and energy

2. Flexible VM relocation

- Live VM migration
  - Flexible VM relocation with near-zero downtime
- Flexible load balancing
  - Relieving resource bottleneck
- High availability

3. Strong isolation

- Strong isolation between co-located VMs
- Fault containment or isolation
  - Safe from bugs and malicious attacks

4. Multiple OS on a single device

- OS dependency of legacy SW
  - Linux + Windows, Android + iOS
- Different requirements of SW
  - Virtualization for embedded or mobile systems
  - Building secure systems

5. Strong security monitoring

- Security monitoring outside OS

6. Ease of deployment

- Virtual appliance
  - A bundle of OS and applications

7. Flexible testing and debugging

- Building distributed environments on a single machine
- Kernel development and debugging
- VM-based recording and replaying

Disadvantages

- **Upfront costs.** The investment in the virtualization software, and possibly additional hardware might be required to make the virtualization possible. This depends on your existing network. Many businesses have sufficient capacity to accommodate the virtualization without requiring a lot of cash. This obstacle can also be more readily navigated by working with a Managed IT Services provider, who can offset this cost with monthly leasing or purchase plans.
- **Software licensing considerations**. This is becoming less of a problem as more software vendors adapt to the increased adoption of virtualization, but it is important to check with your vendors to clearly understand how they view software use in a virtualized environment.
- **Possible learning curve.** Implementing and managing a virtualized environment will require IT staff with expertise in virtualization. On the user side a typical virtual environment will operate similarly to the non-virtual environment. There are some applications that do not adapt well to the virtualized environment – this is something that your IT staff will need to be aware of and address prior to converting.

Virtual Machines

Virtual machines are “an efficient, isolated duplicate of a real machine”

- Popek and Goldberg

Popek and Goldberg introduced conditions for computer architecture to efficiently support system virtualization.

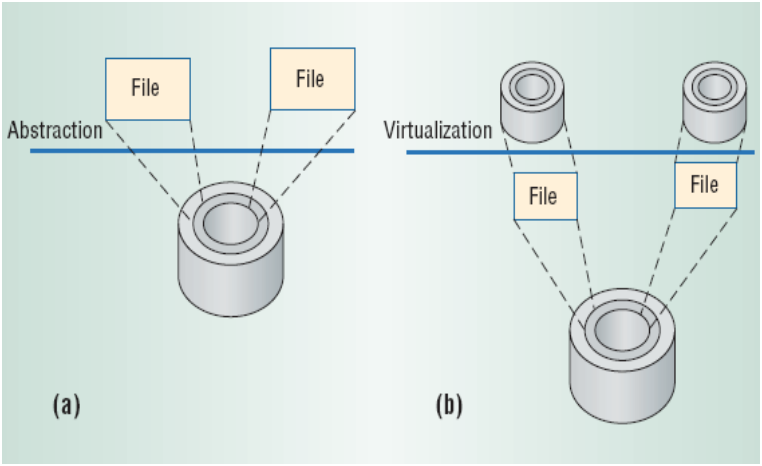
- A **Virtual Machine** is a software that creates a virtualized environment between the computer platform and the end user in which the end user can operate software.

- Concept of *virtualization* applied to the entire machine.
  - mapping of virtual resources or state to real resources
  - use of real machine instructions to carry out actions specified by the virtual machine instructions
- A virtual machine is implemented by adding a layer of software to a real machine to support the desired virtual machine's architecture.
- A virtual machine provides an interface identical to the underlying bare hardware.
- The operating system creates the illusion of multiple processes, each executing on its own processor with its own (virtual) memory.

Abstraction versus Virtualization

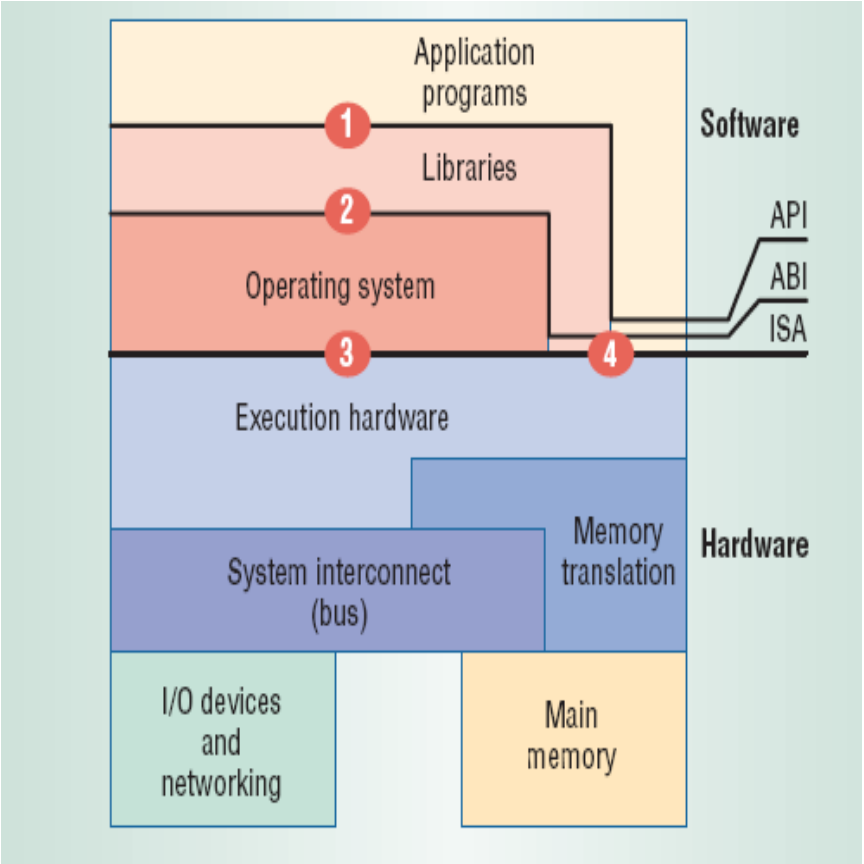
Computer architecture benefits from

- **Abstraction**
  - Well-defined interfaces for hardware and software to use
    - Hard Drives, Networking, I/O devices
  - Limits based on the hardware implementation
- **Virtualization**
  - Maps interfaces and resources to various hardware, even different architectures



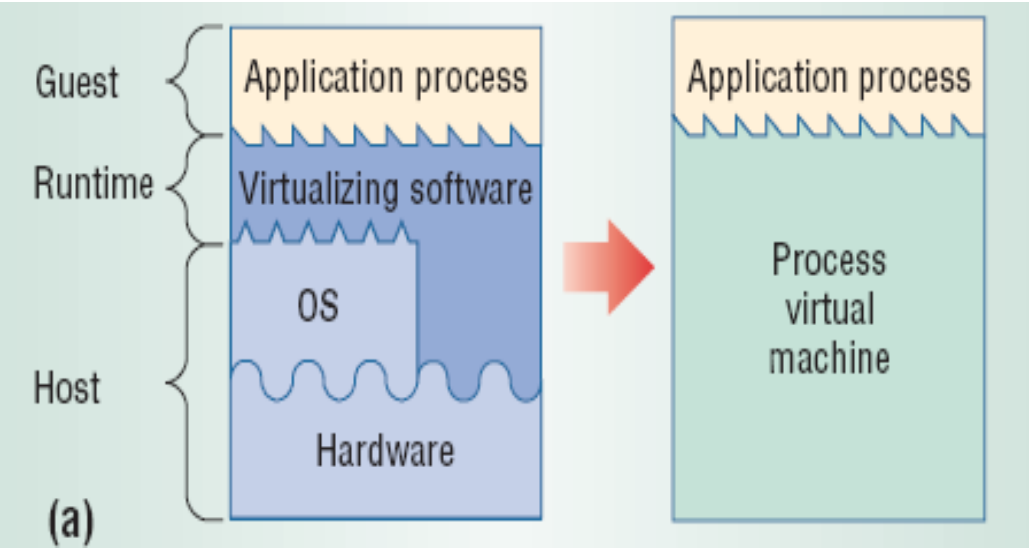
Architecture Interfaces

- Virtualization can take place at these junctures
  - ISA – Instruction Set Architecture
  - ABI – Application Binary Interface
  - API – Application Programming Interface



Process Virtual Machines

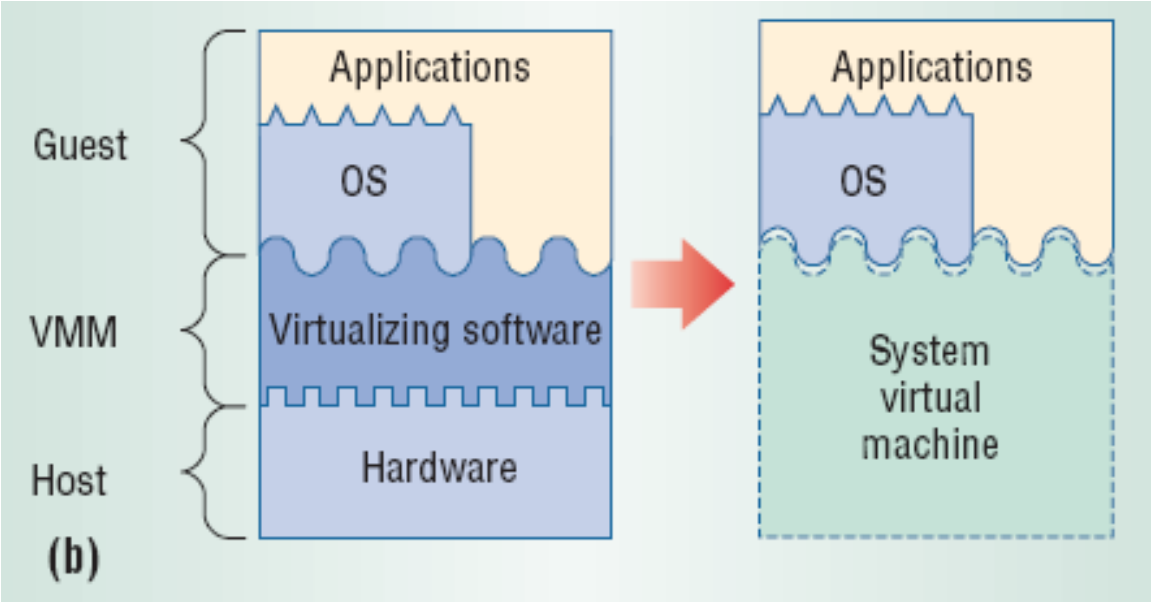
- Also known as Application VM
- Virtualization below the API or ABI, providing virtual resources to a single process executed on a machine
- Created for the process alone, destroyed when process finishes



- **Multiprogrammed Systems**
  - Each application is given effectively separate access to resources, managed by the OS
- **Emulators and Translators**
  - Executes program binaries compiled for different instruction sets.
  - Slower, requiring hardware interpretation
    - Optimization through storing blocks of converted code for repeated execution
- **Optimizers, same ISA**
  - Perform code optimization during translation and execution
- **High-Level-Language VM**
  - Cross-platform compatibility
  - Programs written for an abstract machine, which is mapped to real hardware through a virtual machine
    - Sun Microsystems Java VM
    - Microsoft Common Language Infrastructure, .NET framework.

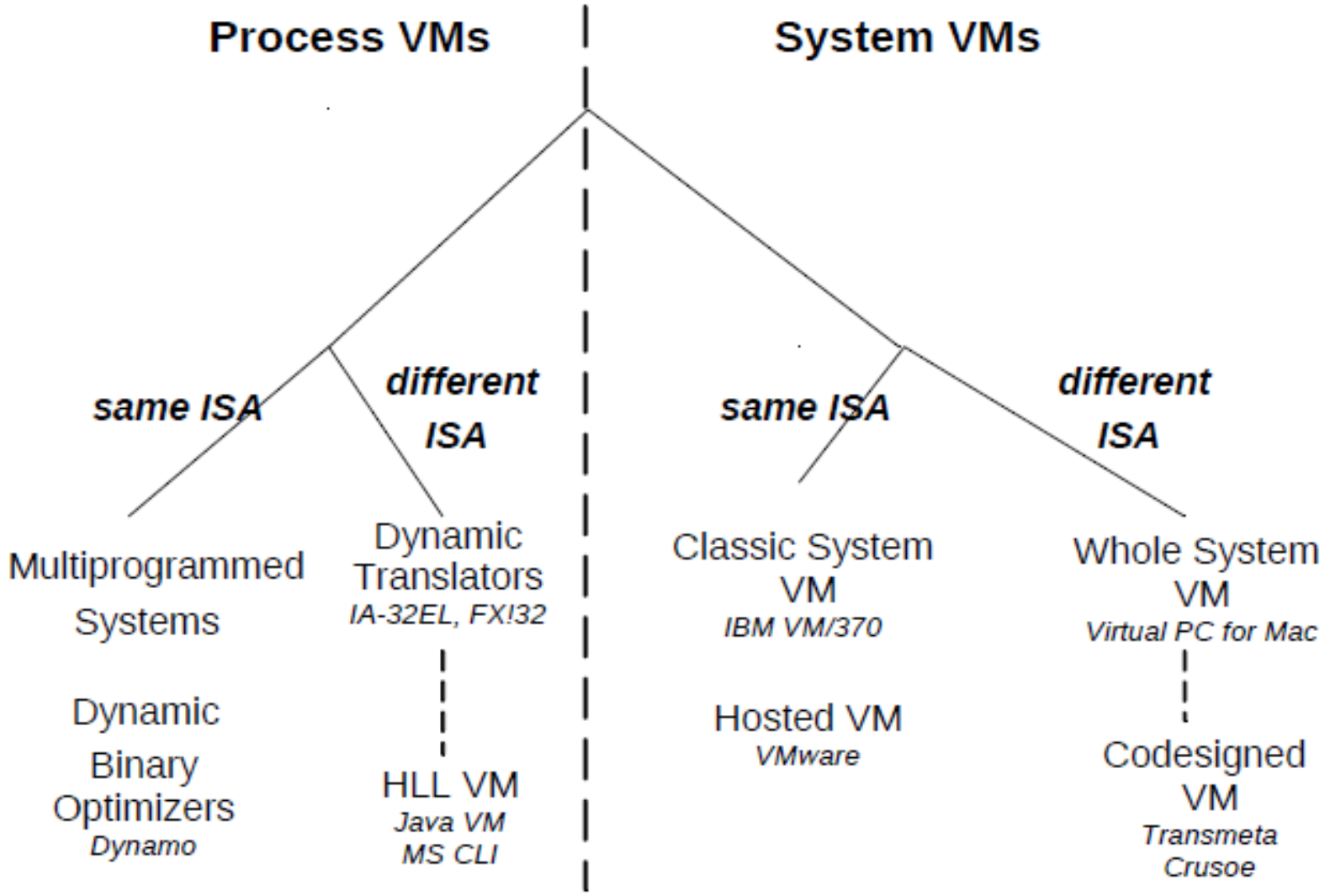
System Virtual Machines

- Virtualized hardware below the ISA
- Single host can run multiple isolated operating systems
  - Servers running different operating systems
  - Isolation between concurrent systems, security
- Hardware Managed by the VMM – Virtual Machine Manager



- Classically, VMM runs on **bare hardware**, directly interacting with resources,
  - Intercepts and interprets guest OS actions
- **Hosted VM**
  - Installed application that relies on the OS to access hardware, using same ISA
- **Whole System VM**
  - Emulate both application and system code for different ISAs
    - Classically: Virtual PC, run windows on old Mac hardware
- **Multiple operating systems on one machine**
  - Processor/resources timeshared
  - Quality-of-service isolation
- **Testing of insecure or questionable software and systems**
  - Better debugging

VM Taxonomy


Process VMs
System VMs
same ISA
different ISA
same ISA
different ISA
Multiprogrammed Systems
Dynamic Binary Optimizers
Dynamo
Dynamic Translators
IA-32EL, FX!32
HLL VM
Java VM
MS CLI
Classic System VM
IBM VM/370
Hosted VM
VMware
Whole System VM
Virtual PC for Mac
Codesigned VM
Transmeta Crusoe

Virtual Machine Properties

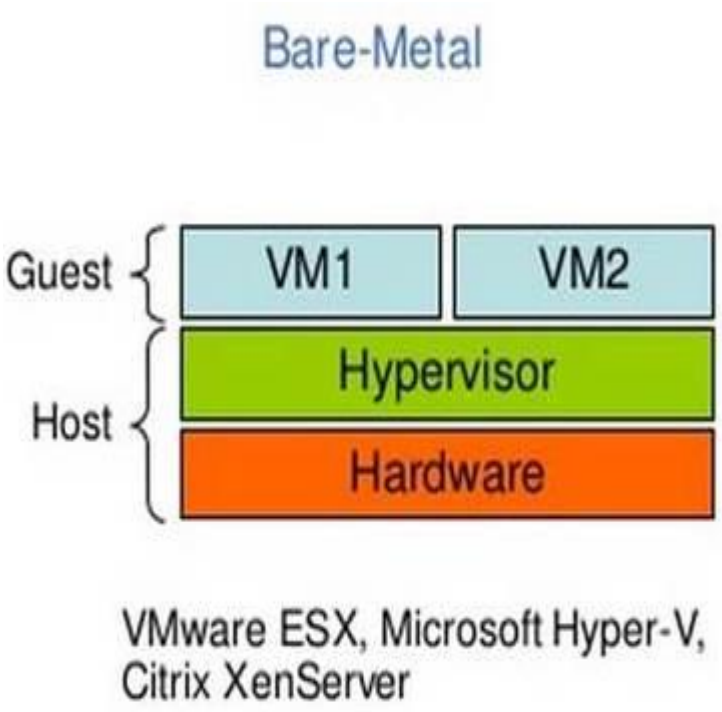
- **Virtual Hardware**
  - Each virtual machine has its own set of virtual hardware (e.g., RAM, CPU, NIC, etc.) upon which an operating system and applications are loaded.
  - The operating system sees a consistent, normalized set of hardware regardless of the actual physical hardware components.
- **Partitioning**
  - Multiple applications and operating systems can be supported within a single physical system.
  - There is no overlap amongst memory as each Virtual Memory has its own memory space.
- **Isolation**
  - Virtual machines are completely isolated from the host machine and other virtual machines. If a virtual machine crashes, all others are unaffected.
  - Data does not leak across virtual machines.
- **Identical Environment**
  - Virtual machines can have a number of discrete identical execution environments on a single computer, each of which runs an operating system.

Virtual Machine Monitors (Hypervisors)

- A **hypervisor** or **virtual machine monitor** (**VMM**) is a piece of computer software, firmware or hardware that creates and runs **virtual machines**.
- A computer on which a hypervisor is running one or more virtual machines is defined as a *host machine*. Each virtual machine is called a *guest machine*
- Hypervisor is a bare metal approach; is installed on the bare metal and then the operating systems is installed (para virtualized)
- Two major types of hypervisor
  - ***Type –I: Bare Metal Hypervisor***
  - ***Type-II: Hosted Hypervisor***

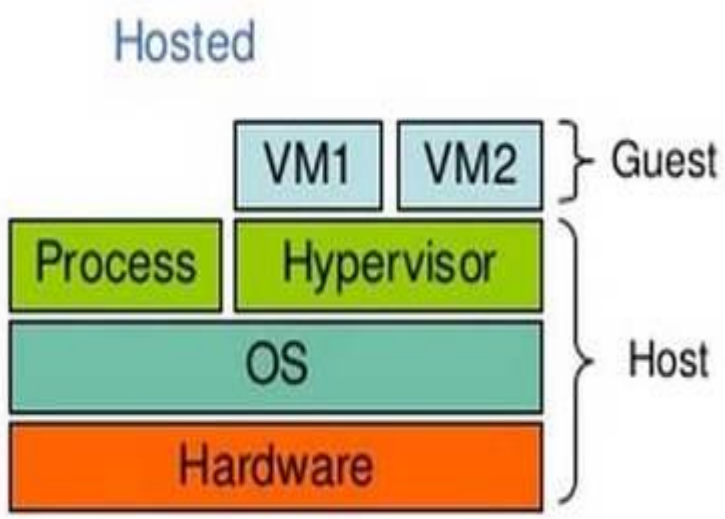
Type-I Hypervisor

- It runs directly on top of the hardware.
- Takes place of OS.
- Directly interact with the ISA exposed by the underlying hardware.
- Also known as *native virtual machine*.



Type-II Hypervisor

- It require the support of an operating system to provide virtualization services.
- Programs managed by the OS.
- Emulate the ISA of virtual h/w.
- Also called hosted virtual machine.



VMware Workstation, Microsoft Virtual PC, Sun VirtualBox, QEMU, KVM

Implementation of VMMs

- Vary greatly, with options including:
  - **Type 0 hypervisors** - Hardware-based solutions that provide support for virtual machine creation and management via firmware
    - IBM LPARs and Oracle LDOMs are examples
  - **Type 1 hypervisors** - Operating-system-like software built to provide virtualization
    - Including VMware ESX, Joyent SmartOS, and Citrix XenServer
  - **Type 1 hypervisors** – Also includes general-purpose operating systems that provide standard functions as well as VMM functions
    - Including Microsoft Windows Server with HyperV and RedHat Linux with KVM
  - **Type 2 hypervisors** - Applications that run on standard operating systems but provide VMM features to guest operating systems
    - Including VMware Workstation and Fusion, Parallels Desktop, and Oracle VirtualBox

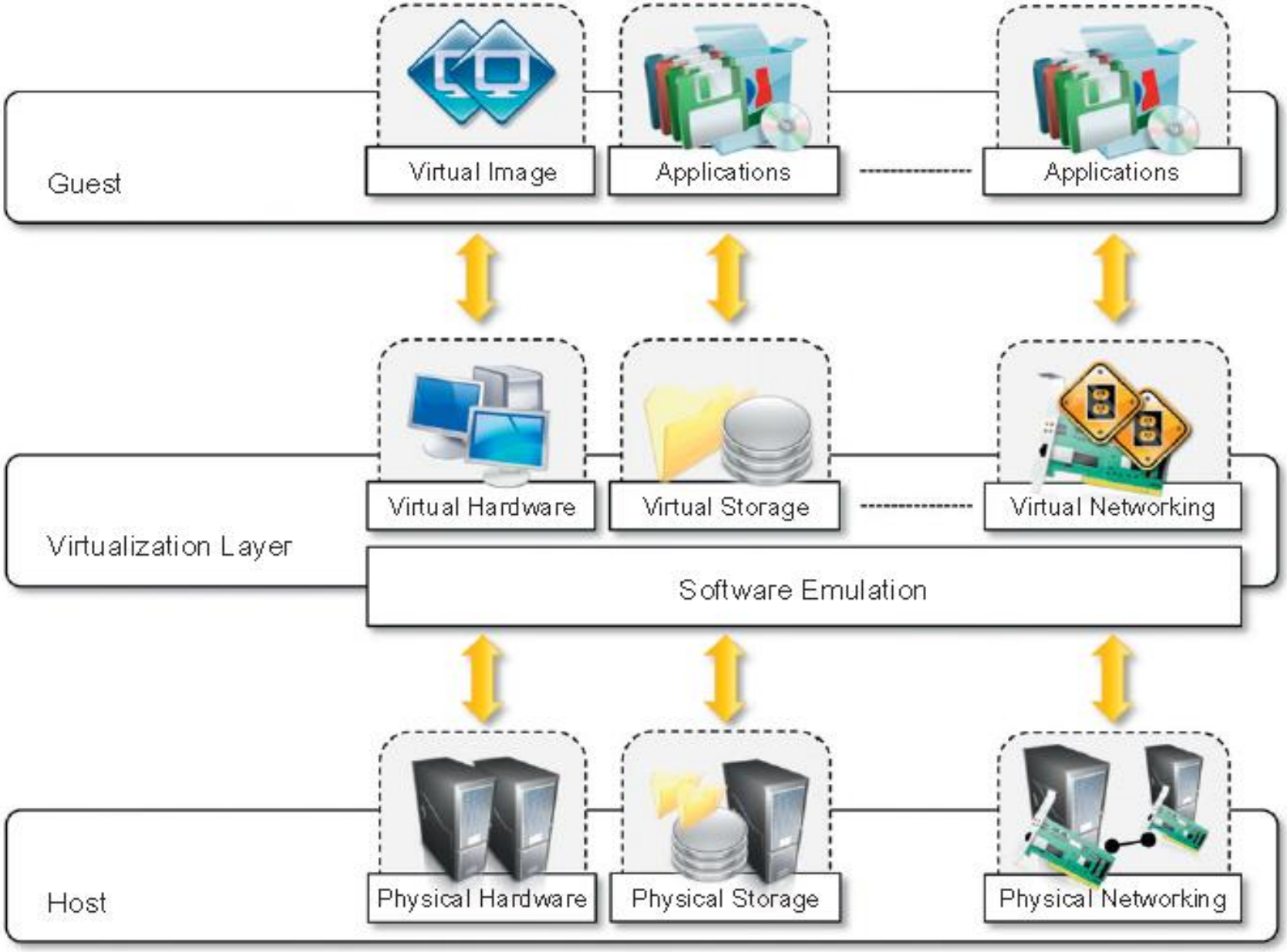
Other variations include:

- **Para virtualization** - Technique in which the guest operating system is modified to work in cooperation with the VMM to optimize performance
- **Programming-environment virtualization** - VMMs do not virtualize real hardware but instead create an optimized virtual system
  - Used by Oracle Java and Microsoft.Net
- **Emulators** – Allow applications written for one hardware environment to run on a very different hardware environment, such as a different type of CPU
- **Application containment** - Not virtualization at all but rather provides virtualization-like features by segregating applications from the operating system, making them more secure, manageable
  - Including Oracle Solaris Zones, BSD Jails, and IBM AIX WPARs
- Much variation due to breadth, depth and importance of virtualization in modern computing

Components of Virtualized Environments

In a virtualized environment there are three major components:

- Guest: The guest represents the system component that interacts with the virtualization layer rather than with the host, as would normally happen.
- Host: The host represents the original environment where the guest is supposed to be managed.
- Virtualization layer: The virtualization layer is responsible for recreating the same or a different environment where the guest will operate


Guest
Virtual Image
Applications
Applications
Virtualization Layer
Virtual Hardware
Virtual Storage
Virtual Networking
Software Emulation
Host
Physical Hardware
Physical Storage
Physical Networking

Characteristics of Virtualized Environments

- **Increased security**
  - The ability to control the execution of a guest in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment.
  - The virtual machine represents an emulated environment in which the guest is executed.
  - All the operations of the guest are generally performed against the virtual machine, which then translates and applies them to the host.
  - By default, the file system exposed by the virtual computer is completely separated from the one of the host machine.
  - This becomes the perfect environment for running applications without affecting other users in the environment.

- **Managed execution**

  Virtualization of the execution environment not only allows increased security, but a wider range of features also can be implemented. The most relevant features are:

  - **Sharing:** Virtualization allows the creation of a separate computing environments within the same host. In this way it is possible to fully exploit the capabilities of a powerful guest, which would otherwise be underutilized.
  - **Aggregation:** A group of separate hosts can be tied together and represented to guests as a single virtual host. This function is naturally implemented in middleware for distributed computing, with a classical example represented by cluster management software, which harnesses the physical resources of a homogeneous group of machines and represents them as a single resource.
  - **Emulation:** Guest programs are executed within an environment that is controlled by the virtualization layer, which ultimately is a program. This allows for controlling and tuning the environment that is exposed to guests.
  - **Isolation:** Virtualization allows providing guests—whether they are operating systems, applications, or other entities—with a completely separate environment, in which they are executed. The guest program performs its activity by interacting with an abstraction layer, which provides access to the underlying resources.

- **Portability**
  - The concept of portability applies in different ways according to the specific type of virtualization considered.
  - In the case of a hardware virtualization solution, the guest is packaged into a virtual image that, in most cases, can be safely moved and executed on top of different virtual machines.
  - In the case of programming-level virtualization, as implemented by the JVM or the .NET runtime, the binary code representing application components (jars or assemblies) can be run without any recompilation on any implementation of the corresponding virtual machine.
  - This makes the application development cycle more flexible and application deployment very straight forward: One version of the application, in most cases, is able to run on different platforms with no changes.
  - Portability allows having your own system always with you and ready to use as long as the required virtual machine manager is available.
  - This requirement is, in general, less stringent than having all the applications and services you need available to you anywhere you go.

Interpretation and Binary Translation

Emulation

- Required for implementing many VMs.
- Process of implementing the interface and functionality of one (sub)system on a (sub)system having a different interface and functionality
  - terminal emulators, such as for VT100, xterm, putty
- Instruction set emulation
  - binaries in *source* instruction set can be executed on machine implementing *target* instruction set. e.g., IA-32 execution layer

Emulation Vs. Simulation

- **Emulation**
  - method for enabling a (sub)system to present the same interface and characteristics as another ways of implementing emulation
    - *interpretation:* relatively inefficient instruction-at-a-time
    - *binary translation:* block-at-a-time optimized for repeated
  - e.g., the execution of programs compiled for instruction set A on a machine that executes instruction set B.
- **Simulation**
  - method for modeling a (sub)system's operation
    - objective is to study the process; not just to imitate the function
    - typically emulation is part of the simulation process

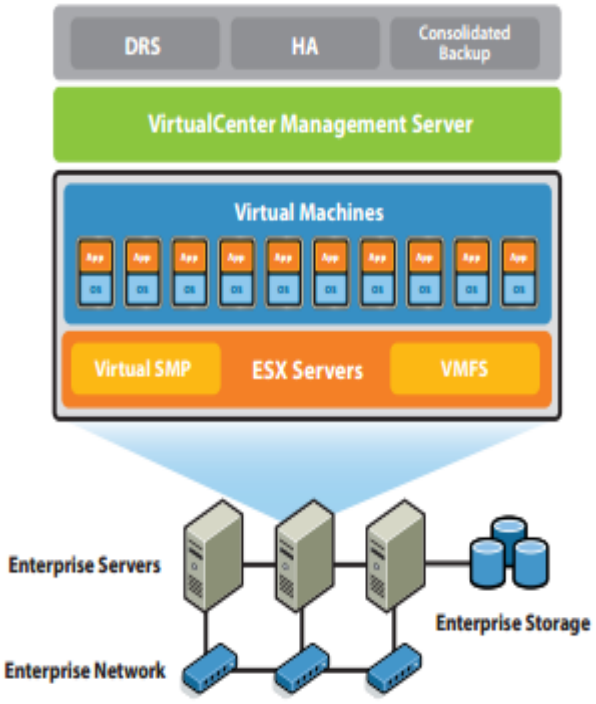
Interpretation Vs. Translation

- **Interpretation**
  1. simple and easy to implement, portable
  2. low performance
  3. threaded interpretation
- **Binary translation**
  1. complex implementation
  2. high initial translation cost, small execution cost
  3. selective compilation

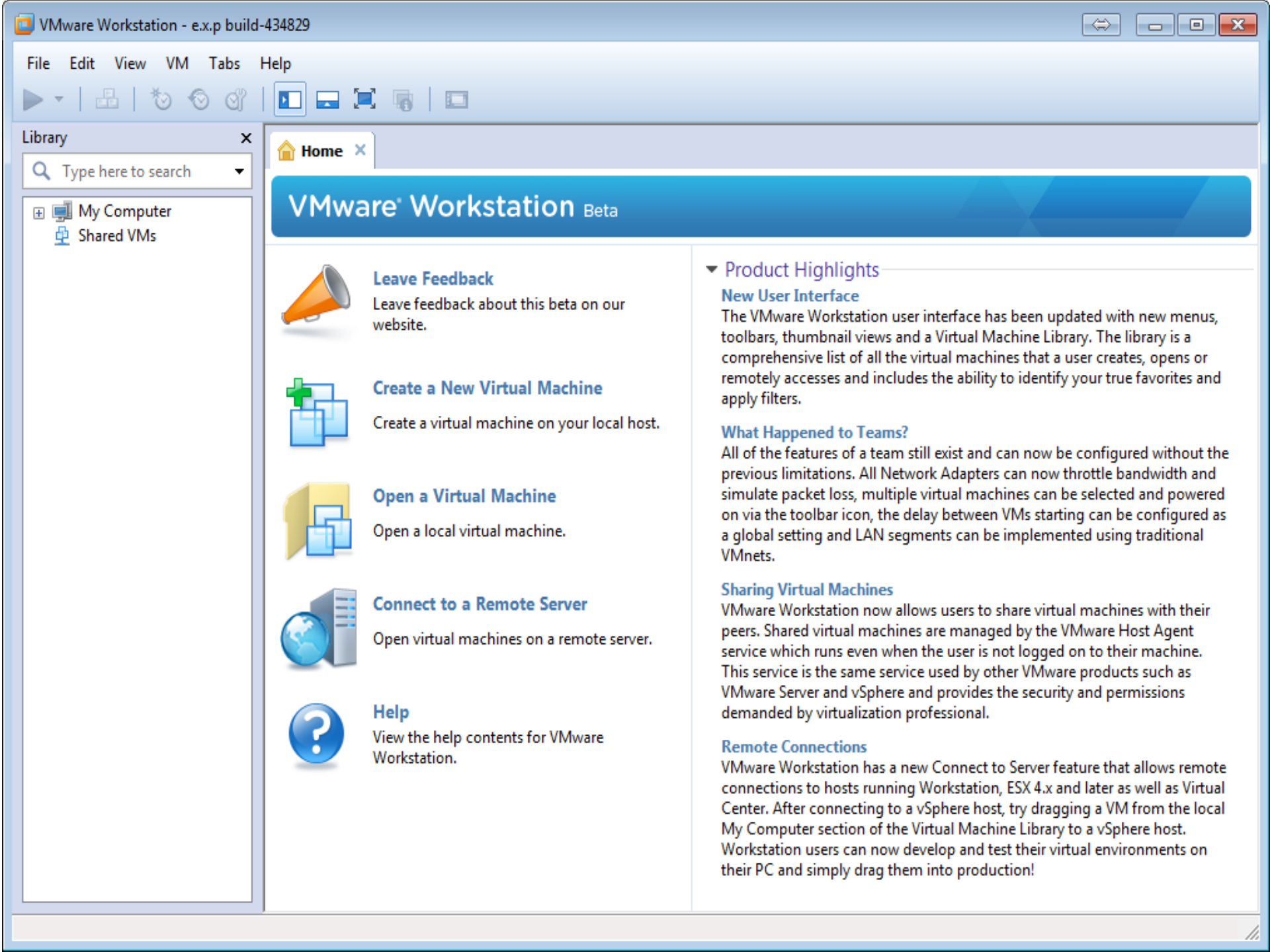
We focus on user-level instruction set emulation of program binaries.

HLL -VM

- Vmware
- Xen
- Hyper-V



- VMware Workstation is the most dependable, high-performing, feature-rich virtualization platform for your Windows or Linux PC.
- It allows one physical PC to run multiple operating systems at the same time.
- No restarting or hard-drive partitioning is required.
- Software developers rely on Workstation to develop and test client-server, Web and cloud applications in a replica of their production environments.


VMware Workstation - e.x.p build-434829
File Edit View VM Tabs Help
Library
Type here to search
My Computer
Shared VMs
Home
VMware Workstation Beta
Leave Feedback
Leave feedback about this beta on our website.
Create a New Virtual Machine
Create a virtual machine on your local host.
Open a Virtual Machine
Open a local virtual machine.
Connect to a Remote Server
Open virtual machines on a remote server.
Help
View the help contents for VMware Workstation.
Product Highlights
New User Interface
The VMware Workstation user interface has been updated with new menus, toolbars, thumbnail views and a Virtual Machine Library. The library is a comprehensive list of all the virtual machines that a user creates, opens or remotely accesses and includes the ability to identify your true favorites and apply filters.
What Happened to Teams?
All of the features of a team still exist and can now be configured without the previous limitations. All Network Adapters can now throttle bandwidth and simulate packet loss, multiple virtual machines can be selected and powered on via the toolbar icon, the delay between VMs starting can be configured as a global setting and LAN segments can be implemented using traditional VMnets.
Sharing Virtual Machines
VMware Workstation now allows users to share virtual machines with their peers. Shared virtual machines are managed by the VMware Host Agent service which runs even when the user is not logged on to their machine. This service is the same service used by other VMware products such as VMware Server and vSphere and provides the security and permissions demanded by virtualization professional.
Remote Connections
VMware Workstation has a new Connect to Server feature that allows remote connections to hosts running Workstation, ESX 4.x and later as well as Virtual Center. After connecting to a vSphere host, try dragging a VM from the local My Computer section of the Virtual Machine Library to a vSphere host. Workstation users can now develop and test their virtual environments on their PC and simply drag them into production!

Key Benefits of VMWare

- Access anytime, anywhere
- Run applications in Windows, Linux and other systems at the same time without restarting.
- Remotely access virtual machines running on Vmware.
- Run as a server to host applications for your team, department or anyone in your organization.
- Create virtual machines that are encrypted, block USB devices and have read-only settings.

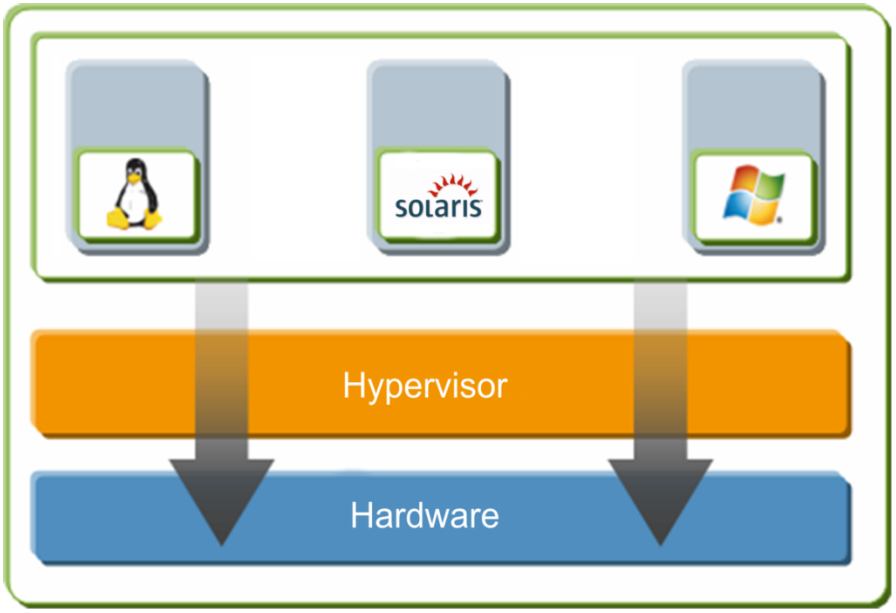
- Xen is a virtual machine monitor for IA-32 (x86, x86-64), IA-64 and PowerPC 970 architectures. It allows several guest operating systems to be executed on the same computer hardware concurrently.
- Xen was initially created by the University of Cambridge, Computer Laboratory and is now developed and maintained by the Xen community as free software, as well as Citrix XenServer Commercial version variant.
- A central part of Amazon.com's cloud computing platform, EC2 allows users to rent virtual computers on which to run their own computer applications.

Xen Architecture

Virtual machine layer

Hypervisor layer

Hardware/physical layer



Hardware or physical layer:

Physical hardware components including memory, CPU, network cards, and disk drives.

Hypervisor layer:

Thin layer of software that runs on top of the hardware. The Xen hypervisor gives each virtual machine a dedicated view of the hardware.

Virtual machine layer:

Operating system hosted on the hypervisor and appearing to the user as a separate physical computer. However, the machine shares physical resources with other virtual machines, and it is portable because the virtual machine is abstracted from the physical hardware.

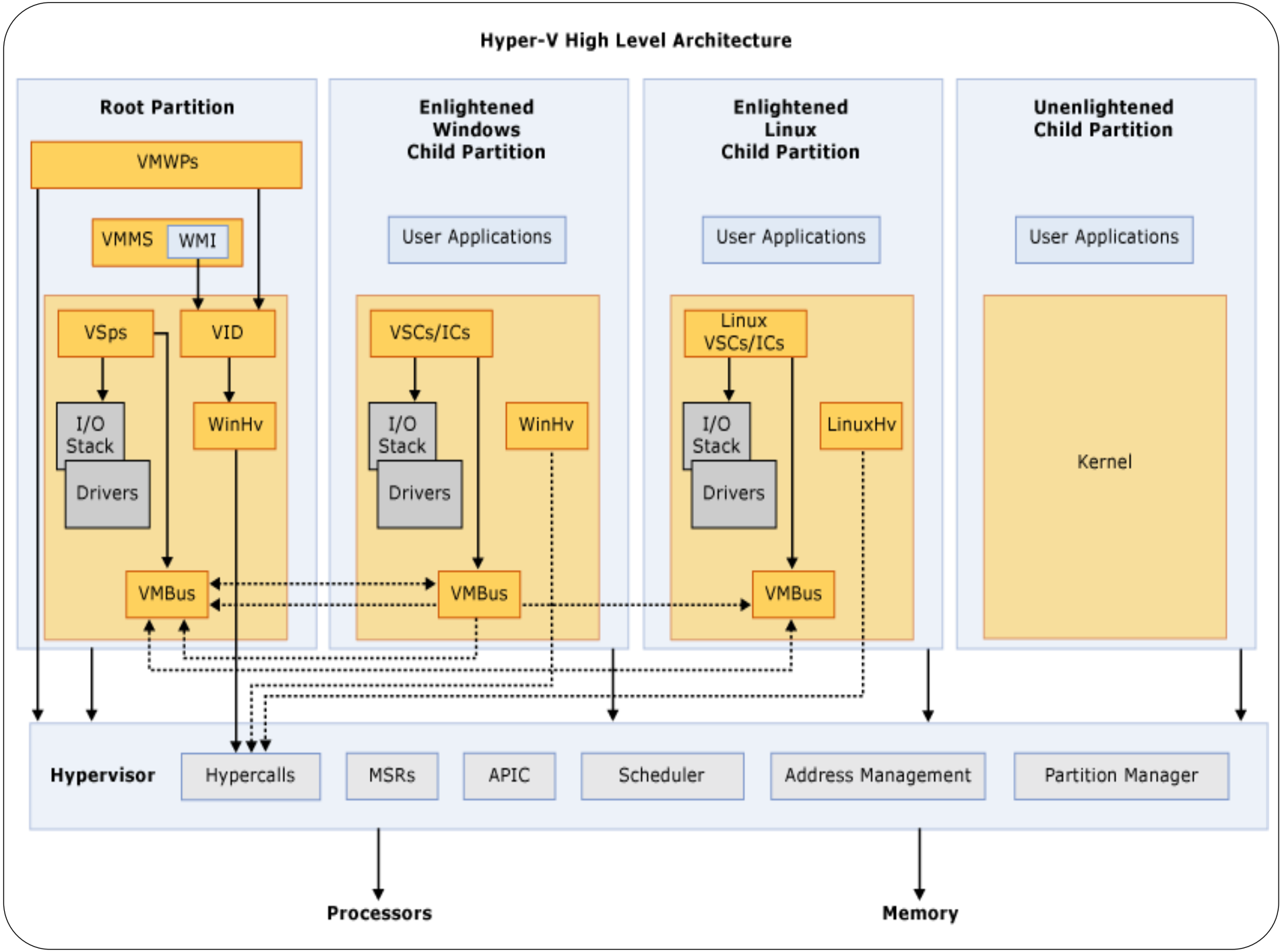
Hyper-V

- Hyper-V is the primary engine that drives Windows Server 2008 "beyond virtualization" initiative.
- The primary responsibility of Windows Server 2008 Hyper-V is to provide the tool kit that organizations will use to create a shared pool of compute, network, and storage resources where servers and applications can be virtualized for consolidation, scalability, and mobility purposes.

Hyper-V Architecture

- Hyper-V is a hypervisor-based virtualization platform and an enabling technology for one of Windows Server 2008 R2's marquee features, Live Migration.
- Guest operating systems running in a Hyper-V virtual machine provide performance approaching the performance of an operating system running on physical hardware *if* the necessary virtual server client (VSC) drivers and services are installed on the guest operating system.
- Hyper-V virtual server client (VSC) code, also known as Hyper-V enlightened I/O, enables direct access to the Hyper-V "Virtual Machine Bus" and is available with the installation of Hyper-V integration services.

- Both Windows Server 2008 R2 and Windows 7 support Hyper-V enlightened I/O with Hyper-V integration services.
- Hyper-V supports isolation in terms of a partition. The Microsoft hypervisor must have at least one parent, or root, partition, running Windows Server 2008 R2.
- The root partition then creates the child partitions which host the guest operating systems.
- Child partitions also do not have direct access to other hardware resources and are presented a virtual view of the resources, as virtual devices (VDevs).


Hyper-V High Level Architecture
Root Partition
VMWPs
VMMS
WMI
VSps
VID
I/O Stack
Drivers
WinHv
VMBus
Enlightened Windows Child Partition
User Applications
VSCs/ICs
I/O Stack
Drivers
WinHv
VMBus
Enlightened Linux Child Partition
User Applications
Linux VSCs/ICs
I/O Stack
Drivers
LinuxHv
VMBus
Unenlightened Child Partition
User Applications
Kernel
Hypervisor
Hypercalls
MSRs
APIC
Scheduler
Address Management
Partition Manager
Processors
Memory

Hyper-V Performance Characteristics

- **Improved hardware sharing architecture** - Hyper-V provides improved access and utilization of core resources, such as disk, networking, and video when running guest operating systems with a hypervisor-aware kernel and which are equipped with requisite virtual server client (VSC) code (known as Hyper-V enlightened I/O).
- **Critical disk performance for I/O intensive applications -** Disk performance is critical for disk I/O intensive enterprise applications such as Microsoft BizTalk Server and in addition to Hyper-V enlightened I/O; Hyper-V provides "Passthrough" disk support which provides disk performance on par with physical disk performance.
- **Processor hardware-assisted virtualization support** – Hyper-V takes full advantage of processor hardware assisted virtualization support that is available with recent processor technology.
- **Multi-core (SMP) guest operating system support** – Hyper-V provides the ability to support up to four processors in a virtual machine environment, which allows applications to take full advantage of multi-threading functionality in a virtual machine.
- **Both 32-bit and 64-bit guest operating system support** – Hyper-V provides broad support for simultaneously running different types of operating systems, including 32-bit and 64-bit systems across different server platforms, such as Windows, Linux, and others.

Advantages of Hyper-V

- **Consolidation of hardware resources** - Multiple physical servers can be easily consolidated into comparatively fewer servers by implementing virtualization with Hyper-V. Consolidation accommodates full use of deployed hardware resources. Hyper-V in Windows Server 2008 R2 can now access up to 64 logical CPUs on host computers.
- **Ease of administration**:
  - Consolidation and centralization of resources simplifies administration.
  - Implementation of scale-up and scale out is accommodated with much greater ease.
- **Significant cost savings**:
  - Hardware costs are significantly reduced because multiple virtual machines can run on a single physical machine
  - Hyper-V licensing costs are included with the license cost of Windows Server 2008 R2.
  - Power requirements may be significantly reduced by consolidating existing applications onto a virtualized Hyper-V environment due to the reduced physical hardware "footprint" that is required.

- **Fault tolerance support through Hyper-V clustering** – Because Hyper-V is a cluster aware application, Windows Server 2008 SP2 provides native host clustering support for virtual machines created in a Hyper-V virtualized environment.
- **Ease of deployment and management**:
  - Consolidation of existing servers into fewer physical servers simplifies deployment.
  - A comprehensive Hyper-V management solution is available with System Center Virtual Machine Manager.
- **Proven track record** - Key Microsoft Web sites MSDN (http://msdn.microsoft.com) and TechNet (http://technet.microsoft.com) are hosted in Hyper-V environments.
- **Comprehensive product support** – Because Microsoft enterprise applications (such as Exchange Server and SQL Server) are fully tested running in Hyper-V, Microsoft provides code fix support for these applications when deployed and run in a Hyper-V environment.
- **Scalability** – Additional processing power, network bandwidth, and storage capacity can be accomplished quickly and easily by apportioning additional available resources from the host computer to the guest virtual machine(s).

Disadvantages of Hyper-V

- **Hardware requirements –** Due to the demands of server consolidation, Hyper-V virtual machines tend to
  - consume more CPU and memory, and
  - require greater disk I/O bandwidth than physical servers with comparable computing loads.
  - Because the Hyper-V server role is only available for 64-bit and all editions of Windows Server 2008 R2 are 64-bit only, the physical hardware must support hardware assisted virtualization. This means the processor must be compatible with Intel VT or AMD Virtualization (AMD-V) technology, the system BIOS must support Data Execution Prevention (DEP), and DEP must be enabled.
- **Software requirements –** While most Microsoft software is supported running on Hyper-V virtual machines,
  - some Microsoft software is still in the process of being tested to ensure compatibility with a Hyper-V virtualized environment.
    - For example, most Microsoft enterprise level applications either support running on Hyper-V or are in the process of being tested for support on Hyper-V. All versions of BizTalk Server since BizTalk Server 2004 are supported running on Hyper-V.

Taxonomy of Virtualisation Techniques