Virtualization and Cloud Computing

By: Neelam Singh

Definition

- Virtualization is the ability to run multiple operating systems on a single physical system and share the underlying hardware resources*
- It is the process by which one computer hosts the appearance of many computers.
- Virtualization is used to improve IT throughput and costs by using physical resources as a pool from which virtual resources can be allocated.

*VMWare white paper, *Virtualization Overview*

Virtualization & Cloud Computing

- **Virtualization:**

  *The ability to run multiple operating systems on a single physical system and share the underlying hardware resources**

- **Cloud Computing:**

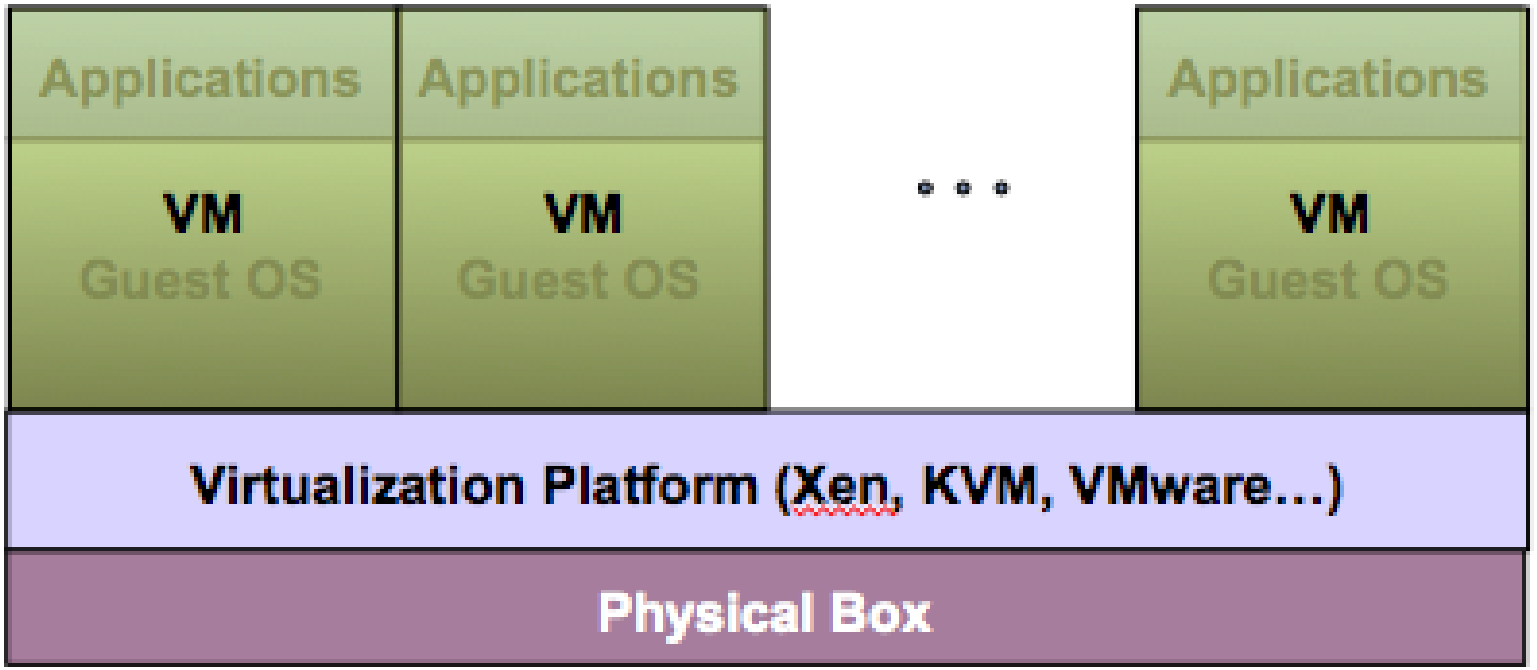
  *"The provisioning of services in a timely (near on instant), on-demand manner, to allow the scaling up and down of resources"***

* VMware white paper, *Virtualization Overview*

** Alan Williamson, quoted in *Cloud BootCamp March 2009*

Virtualization Architecture

- A Virtual machine (VM) is an isolated runtime environment (guest OS and applications)
- Multiple virtual systems (VMs) can run on a single physical system



Hypervisor

- A hypervisor, a.k.a. a virtual machine manager/monitor (VMM), or virtualization manager, is a program that allows multiple operating systems to share a single hardware host.
- Each guest operating system appears to have the host's processor, memory, and other resources all to itself. However, the hypervisor is actually controlling the host processor and resources, allocating what is needed to each operating system in turn and making sure that the guest operating systems (called virtual machines) cannot disrupt each other.

Benefits of Virtualization

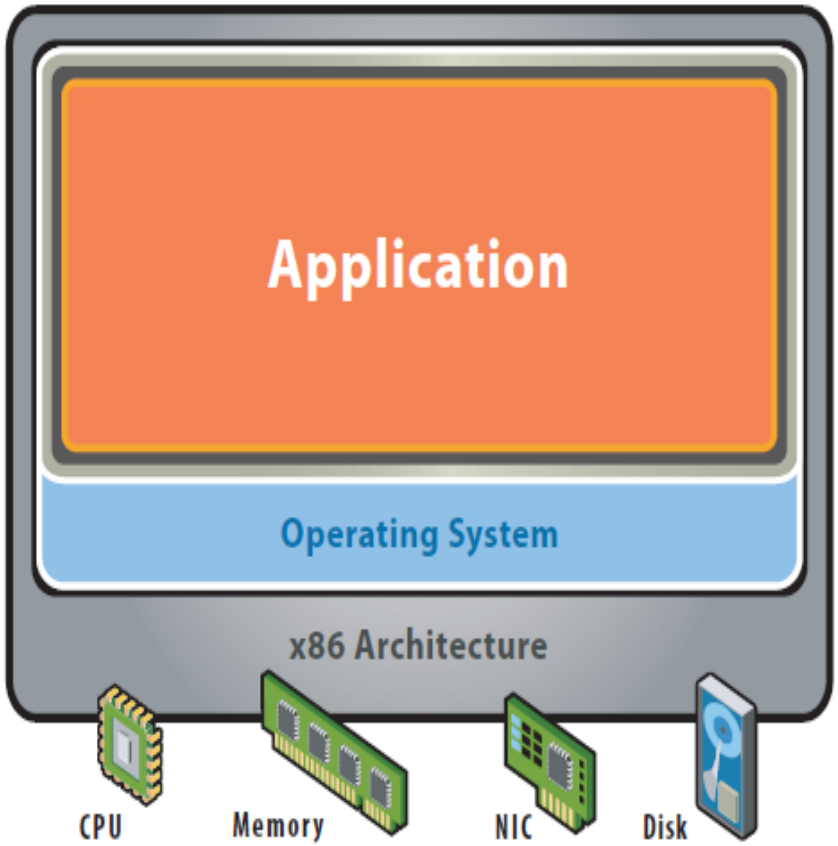
- Sharing of resources helps cost reduction
- Isolation: Virtual machines are isolated from each other as if they are physically separated
- Encapsulation: Virtual machines encapsulate a complete computing environment
- Hardware Independence: Virtual machines run independently of underlying hardware
- Portability: Virtual machines can be migrated between different hosts.

Virtualization in Cloud Computing

Cloud computing takes virtualization one step further:

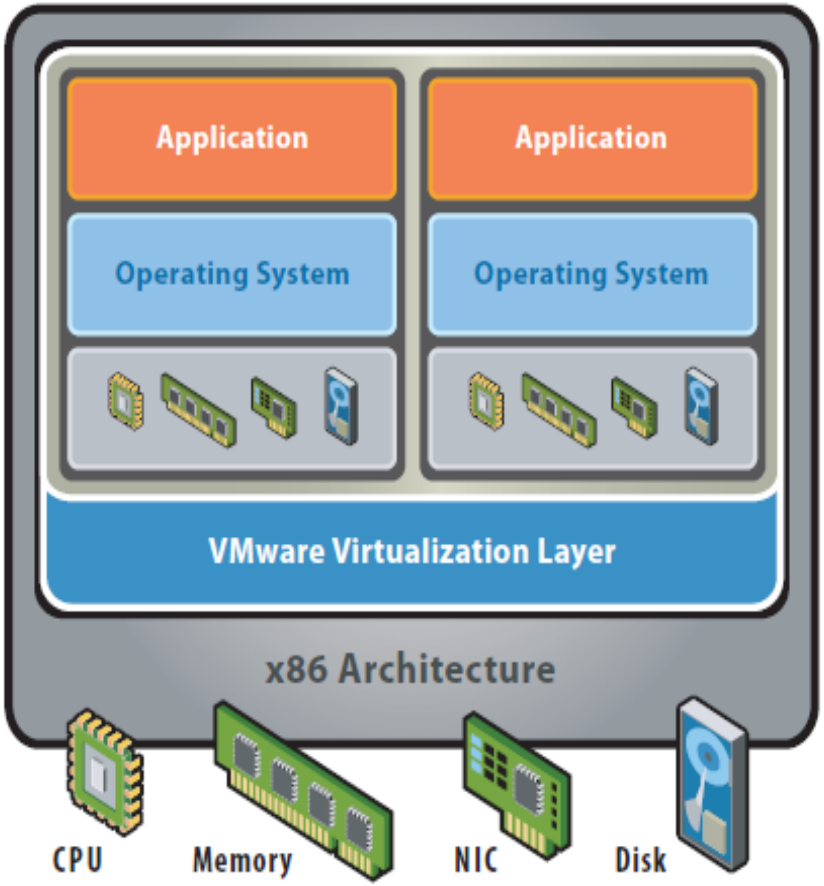
- You don’t need to own the hardware
- Resources are rented as needed from a cloud
- Various providers allow creating virtual servers:
  - Choose the OS and software each instance will have
  - The chosen OS will run on a large server farm
  - Can instantiate more virtual servers or shut down existing ones within minutes
- You get billed only for what you used

Before Virtualization



- Single OS image per machine
- Software and hardware tightly coupled
- Running multiple applications on same machine often creates conflict
- 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

Types of Virtualization

- Virtual memory
- Desktop virtualization
- Platform virtualization
  - Full virtualization
  - Paravirtualization
  - Hardware-assisted virtualization
  - Partial virtualization
  - OS-level virtualization
  - Hosted environment (e.g. User-mode Linux)
- Storage virtualization
- Network virtualization
- Application virtualizationPortable application
  - Cross-platform virtualization
  - Emulation or simulation
  - Hosted Virtual Desktop

Category in Wiki

- In this talk, we mainly focus on Platform virtualization which is mostly related to cloud-computing
  - Full virtualization
  - Binary transaltion
  - Hardware-assisted virtualization
  - Paravirtualization
  - OS-level virtualization
  - Hosted environment (e.g. User-mode Linux)

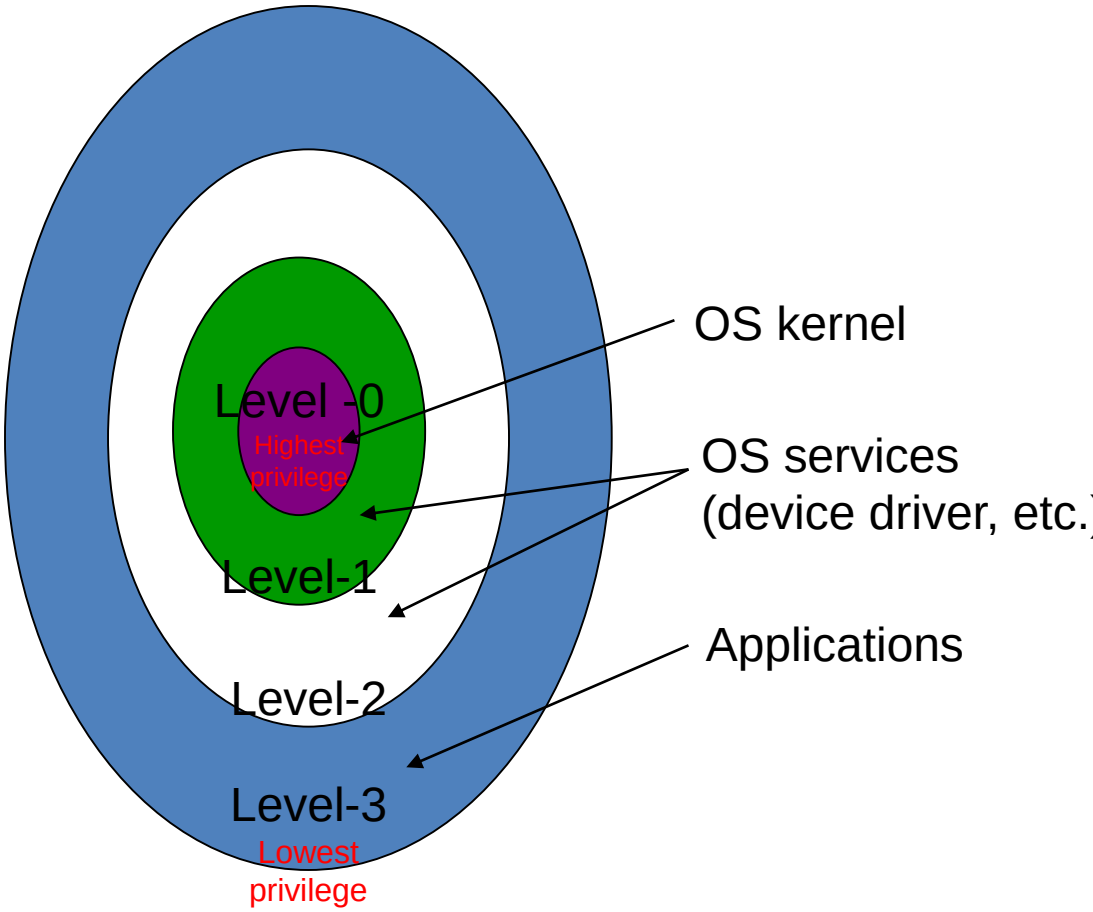
  - Hardware level
  - Operating system level
  - Application level

Platform Virtualization

- *Hardware virtualization* or *platform virtualization* refers to the creation of a virtual machine that acts like a real computer with an operating system. Software executed on these virtual machines is separated from the underlying hardware resources. For example, a computer that is running Microsoft Windows may host a virtual machine that looks like a computer with the Ubuntu Linux operating system; Ubuntu-based software can be run on the virtual machine.
- In hardware virtualization, the ***host*** *machine* is the actual machine on which the virtualization takes place, and the ***guest*** *machine* is the virtual machine. The words *host* and *guest* are used to distinguish the software that runs on the physical machine from the software that runs on the virtual machine. The software or firmware that creates a virtual machine on the host hardware is called a ***hypervisor*** or *Virtual Machine Manager*.

x86 Hardware Virtualization

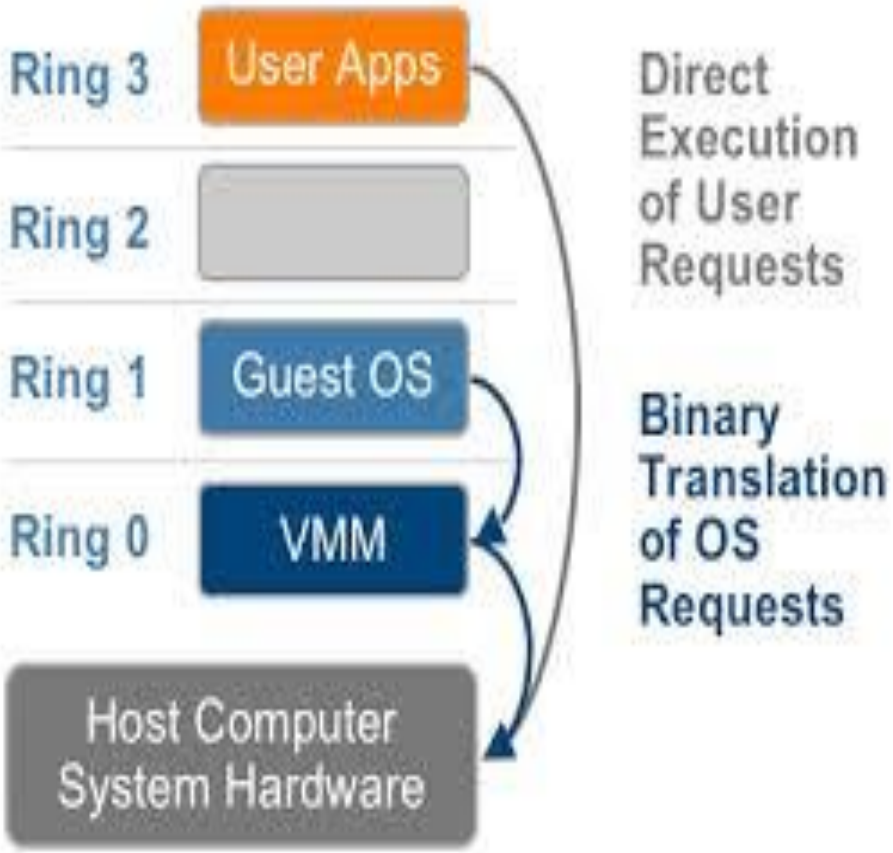
- The x86 architecture offers four levels of privilege known as Ring 0, 1, 2 and 3 to operating systems and applications to manage access to the computer hardware. While user level applications typically run in Ring 3, the operating system needs to have direct access to the memory and hardware and must execute its privileged instructions in Ring 0.



Technique 1. Full Virtualization

•**Full virtualization** is a virtualization technique used to provide a certain kind of virtual machine environment, namely, one that is a complete simulation of the underlying hardware.

•This approach relies on binary translation to trap (into the VMM) and to virtualize certain sensitive and non-virtualizable instructions with new sequences of instructions that have the intended effect on the virtual hardware. Meanwhile, user level code is directly executed on the processor for high performance virtualization.



Binary translation approach to x86 virtualization

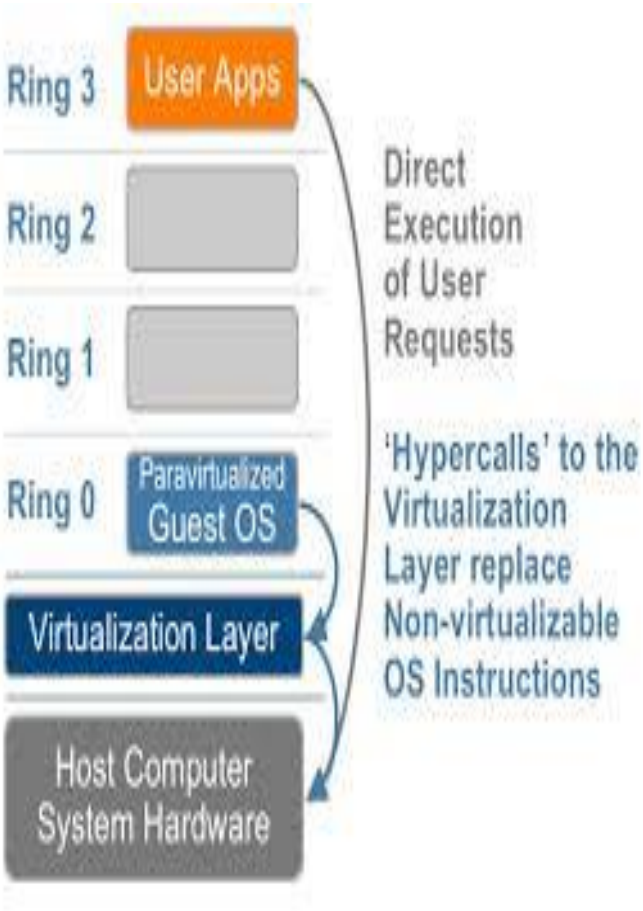
Full Virtualization

- This combination of binary translation and direct execution provides Full Virtualization as the guest OS is completely decoupled from the underlying hardware by the virtualization layer.
- The guest OS is not aware it is being virtualized and requires no modification.
- The hypervisor translates all operating system instructions at run-time on the fly and caches the results for future use, while user level instructions run unmodified at native speed.
- VMware's virtualization products such as VMWare ESXi and Microsoft Virtual Server are examples of full virtualization.

Advantages of Full Virtualization

- Full virtualization technique grants the potential to combine existing systems on to the newer ones with increased efficiency and a well-organized hardware.
- This amazing methodology contributes effectively to trim down the operating costs engaged in repairing and enhancing older systems.
- The less competent systems can be power-packed with this technique, while reducing the physical space and augmenting the overall performance of the company.
- Full virtualization has proven highly successful for:
  - sharing a computer system among multiple users;
  - isolating users from each other (and from the control program);
  - emulating new hardware to achieve improved reliability, security and productivity

Technique 2: ParaVirtualization or OS Assisted Virtualization



Paravirtualization approach to x86 Virtualization

- In computing, **paravirtualization** is a virtualization technique that presents a software interface to virtual machines that is similar, but not identical to that of the underlying hardware.
- Paravirtualization refers to communication between the guest OS and the hypervisor to improve performance and efficiency.
- Paravirtualization involves modifying the OS kernel to replace nonvirtualizable instructions with hypercalls that communicate directly with the virtualization layer hypervisor.
- The hypervisor also provides hypercall interfaces for other critical kernel operations such as memory management, interrupt handling and time keeping.

ParaVirtualization or OS Assisted Virtualization

- It is very difficult to build the more sophisticated binary translation support necessary for full virtualization.
- Paravirtualization involves modifying the OS kernel to replace non-virtualizable instructions with hypercalls that communicate directly with the virtualization layer hypervisor.
- The hypervisor also provides hypercall interfaces for other critical kernel operations such as memory management, interrupt handling and time keeping.
- Paravirtualization is different from full virtualization, where the unmodified OS does not know it is virtualized and sensitive OS calls are trapped using binary translation.
- Paravirtualization cannot support unmodified OS

- Example:
  - Xen -- modified Linux kernel and a version of Windows XP

Advantages of Para Virtualization

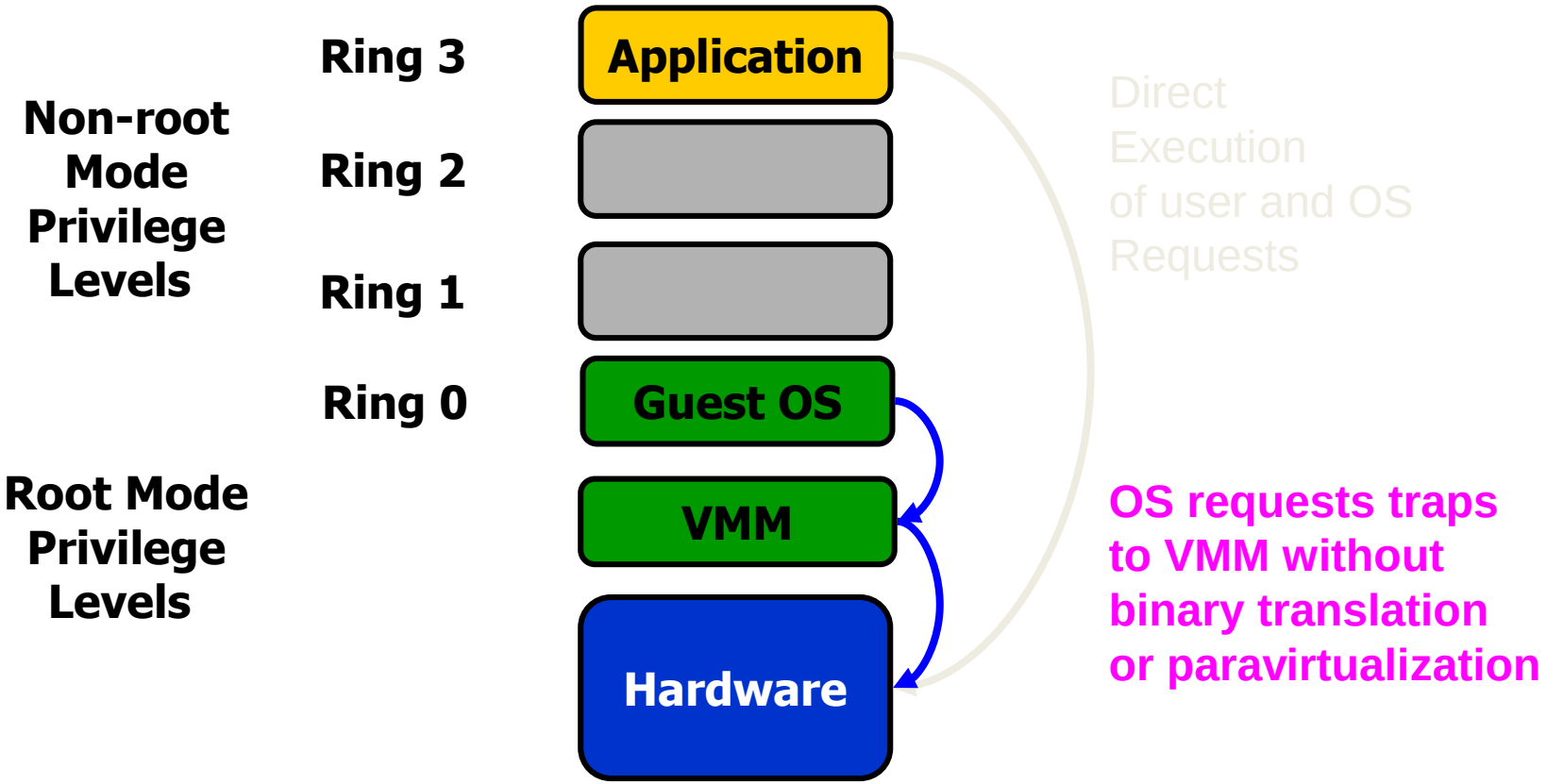
- It enhances the performance notably by decreasing the number of VMM calls and prevents the needless use of privileged instructions.
- It allows running many operating systems on a single server.
- This method is considered as the most advantageous one as it augments the performance per server without the operating cost of a host operating system.
- paravirtualization eliminates the need for the virtual machine to trap privileged instructions. Trapping, a means of handling unexpected or unallowable conditions, can be time-consuming and can adversely impact performance in systems that employ full virtualization.
- **Disadvantage**
  - The major **drawback of paravirtualization** is the requirement of modifying guest operating system to execute and communicate with the hypervisor.

Hardware Assisted Virtualization

- Also known as accelerated virtualization, hardware virtual machine (Xen), native virtualization (Virtual iron).
- Hardware switch supported by CPU, e.g.
  - Intel Virtualization Technology (VT-x)
  - AMD's AMD-V

  target privileged instructions with a new CPU execution mode feature that al lows the VMM to run in a new root mode below ring 0.
- Privileged and sensitive calls are set to automatically trap to the hypervisor, removing the need for either binary translation or paravirtualization.
- The guest state is stored in Virtual Machine Control Structures (VT-x) or Virtual Machine Control Blocks (AMD-V).
- High hypervisor to guest transition overhead and a rigid programming model

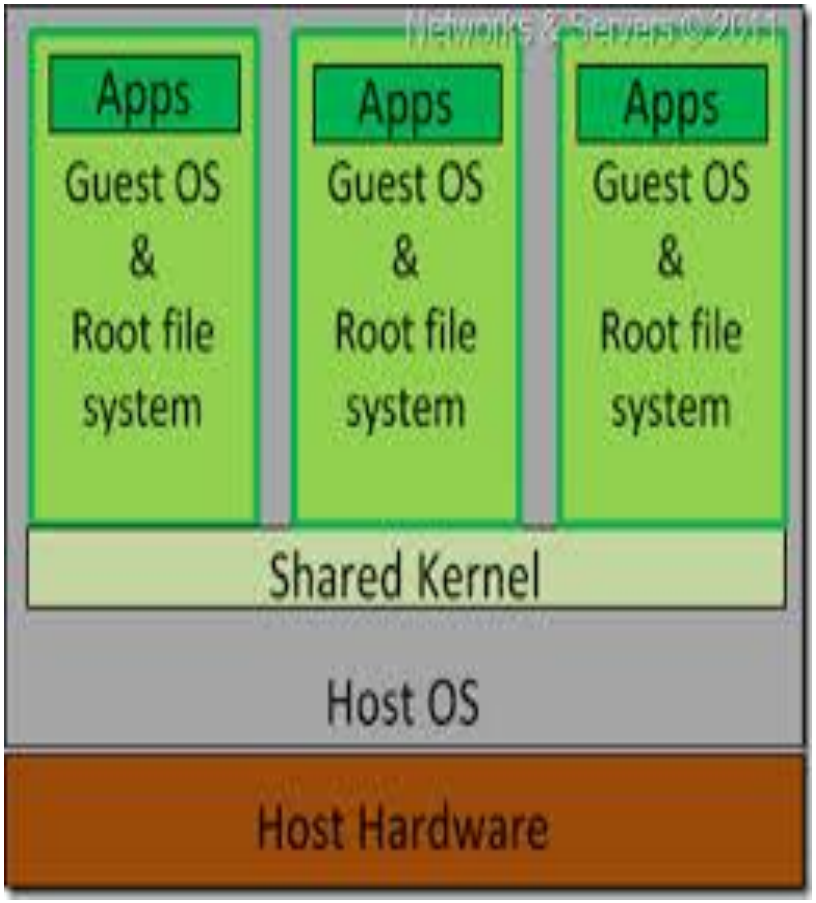
Hardware Assisted Virtualization



VMM: Virtual Machine Monitor

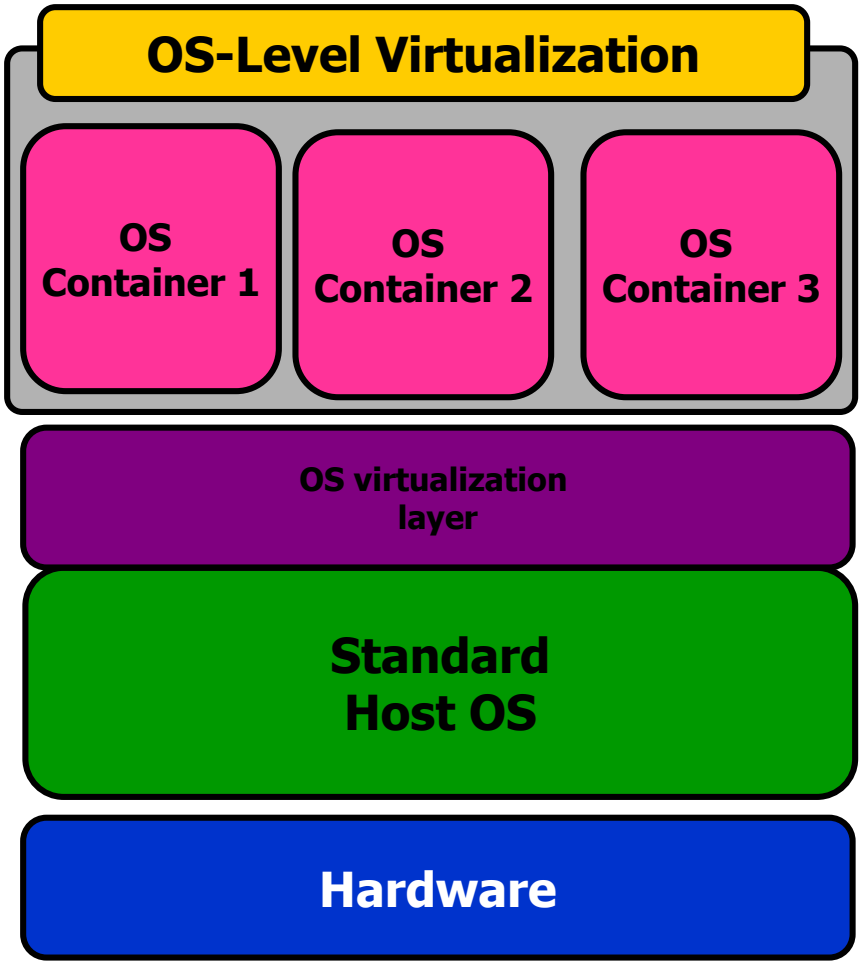
Operating system virtualization

- Operating system virtualization helps to create virtualized layer of software on the top of host operating system that resides above the hardware layer. Unlike other virtualization, they create an OS interfaces for applications to run, giving the feeling of a complete OS for the applications. Each virtualized environment has its own file system, system libraries, process tables and network configuration. Since they create a self-contained environment, they are also known as "container". Therefore, creating the software emulation of an entire OS in a physical server is the essence of OS virtualization.
- **Operating-system-level virtualization** is a server-virtualization method where the kernel of an operating system allows for multiple isolated user-space instances, instead of just one. Such instances (sometimes called **containers**, **software containers**, virtualization engines (VE), virtual private servers (VPS), or jails) may look and feel like a real server from the point of view of its owners and users.



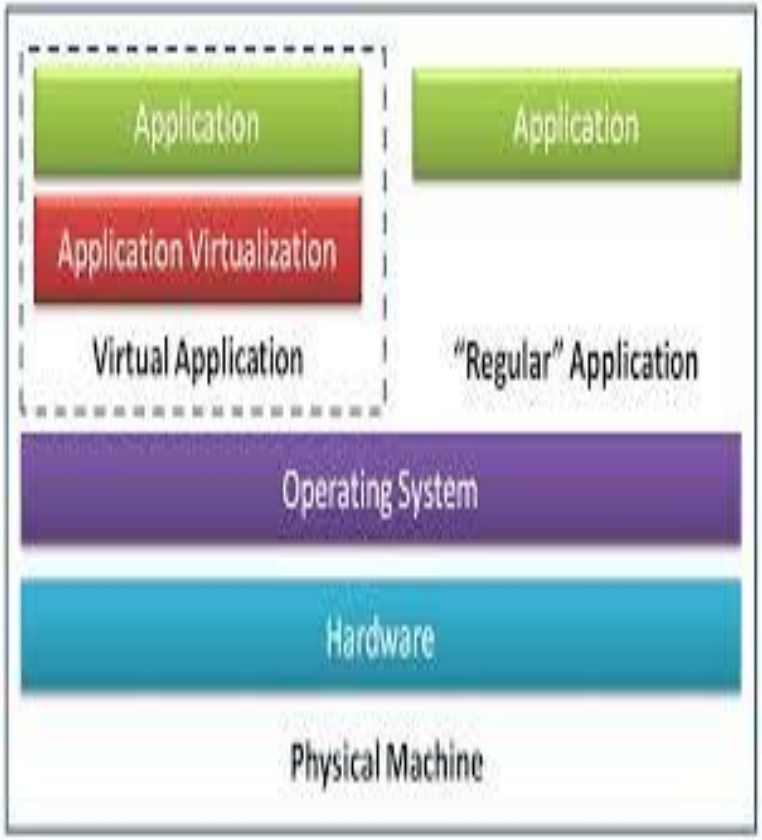
Operating system virtualization

- OS-level virtualization
  - kernel of an OS allows for multiple isolated user-space instances, instead of just one.
  - Each OS instance looks and feels like a real server
- OS virtualization virtualizes servers on the operating system (kernel) layer. This creates isolated containers on a single physical server and OS instance to utilize hardware, software, data center and management efforts with maximum efficiency.
- OS-level virtualization implementations that are capable of live migration can be used for dynamic load balancing of containers between nodes in a cluster.



Application virtualization

- **Application virtualization** is software technology that encapsulates computer programs from the underlying operating system on which it is executed
- Application runs on
  - Different OS, platform, etc.
  - Same OS, different version/framework
  - Encapsulation of OS/platform
  - Improve portability, manageability and compatibility of applications
- A fully virtualized application is not installed in the traditional sense, although it is still executed as if it is (runtime virtualization)
- Full application virtualization requires a virtualization layer.
- Examples of this technology for the Windows platform include:
  - 2X Software
  - Cameyo
  - Ceedo
  - Citrix XenApp
  - InstallFree
  - Microsoft App-V
  - Numecent Application Jukebox
  - Oracle Secure Global Desktop

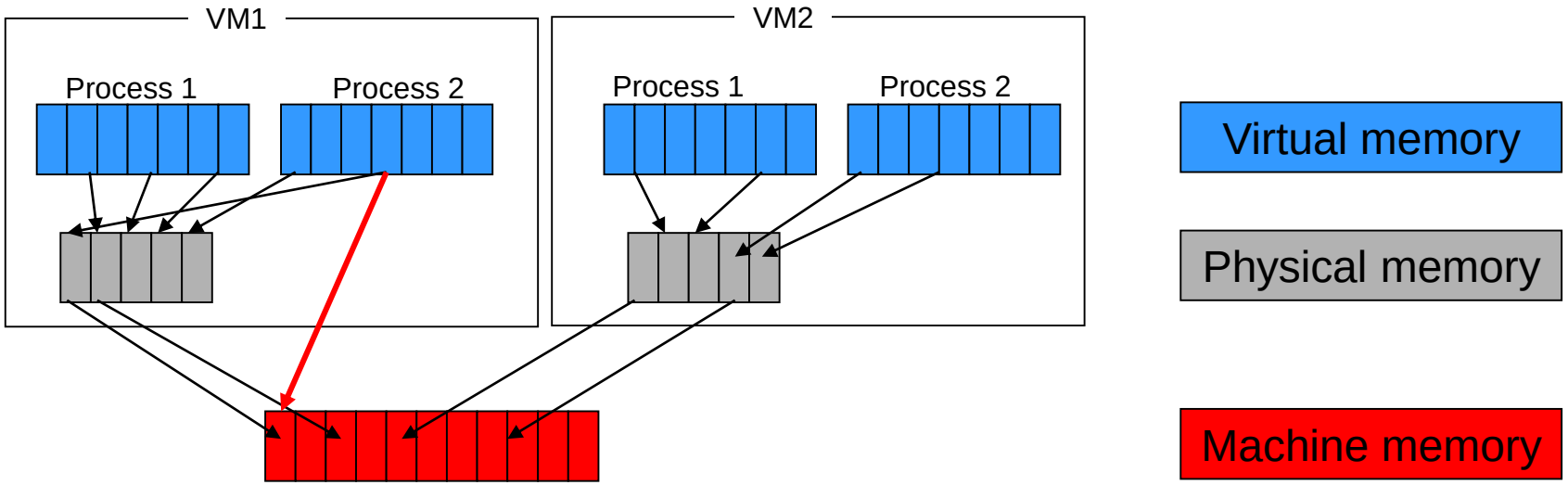


Benefits & Limitations

- Benefits
  - Application virtualization allows applications to run in environments that do not suit the native application. For example, Wine allows some Microsoft Windows applications to run on Linux.
  - Application virtualization reduces system integration and administration costs by maintaining a common software baseline across multiple diverse computers in an organization.
  - Application virtualization also enables simplified operating system migrations.
  - Application virtualization uses fewer resources than a separate virtual machine.
- Limitations
  - Not all computer programs can be virtualized. Some examples include applications that require a device driver (a form of integration with the OS) and 16-bit applications that need to run in shared memory space

Memory Virtualization

- Not only virtual memory
- Hardware support
  - e.g., x86 MMU and TLB
- To run multiple virtual machines on a single system, another level of memory virtualization is required.
- The VMM is responsible for mapping guest physical memory to the actual machine memory, and it uses shadow page tables to accelerate the mappings.



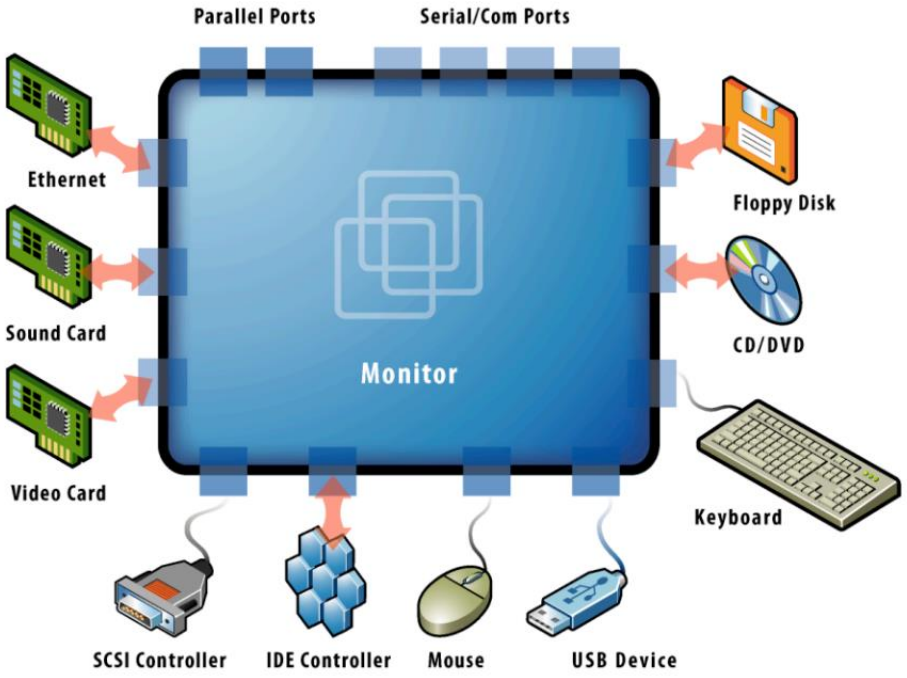
- Memory Virtualization decouples random access memory (RAM) resources from individual systems in the data center, and then aggregates those resources into a virtualized memory pool available to any computer in the cluster.
- The memory pool is accessed by the operating system or applications running on top of the operating system.
- The distributed memory pool can then be utilized as a high-speed cache, a messaging layer, or a large, shared memory resource for a CPU or a GPU application.

Benefits:

- Improves memory utilization via the sharing of scarce resources.
- Increases efficiency and decreases run time for data intensive and I/O bound applications
- Allows applications on multiple servers to share data without replication, decreasing total memory needs
- Lowers latency and provides faster access than other solutions.

Device and I/O Virtualization

- VMM supports all device/IO drivers
- Physically/virtually existed



Source: VMware white paper, “Understanding Full Virtualization, Paravirtualization, and Hardware Assist”

Device and I/O Virtualization

- I/O Virtualization environments are created by abstracting the upper layer protocols from the physical connections.
- One physical adapter card appear as multiple virtual network interface cards (vNICs) and virtual host bus adapters.
- In the physical view, virtual I/O replaces a server's multiple I/O cables with a single cable that provides a shared transport for all network and storage connections.
- Simplify management, lower costs and improve performance of servers in enterprise environments.

Benefits

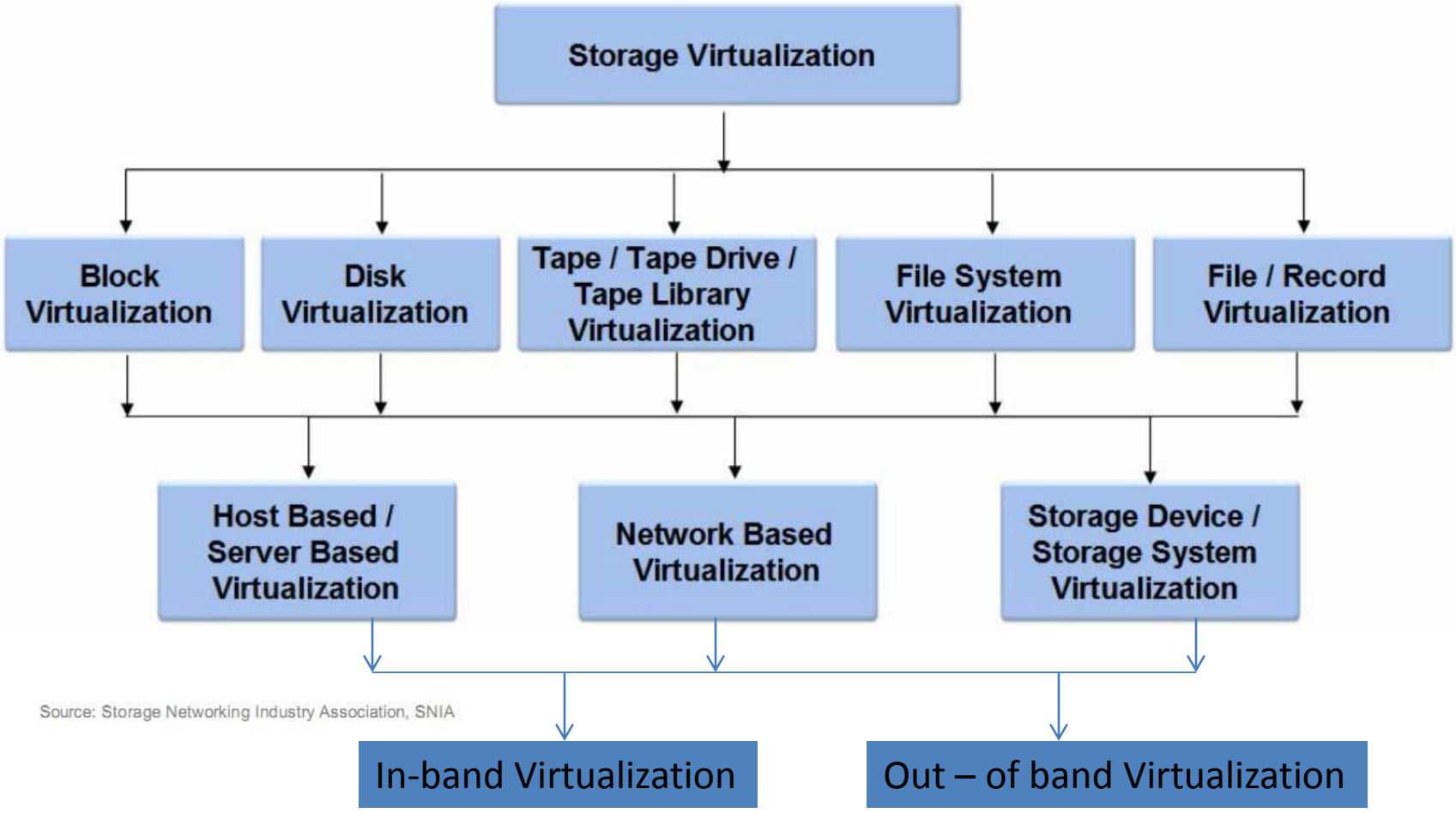
- **•Management agility: By abstracting upper layer protocols from physical connections, I/O Virtualization provides greater flexibility, greater utilization and faster provisioning when compared to traditional architectures.**

- **•Reduced cost*:Virtual I/O lowers costs and enables simplified server management by using fewer cards, cables, and switch ports, while still achieving full network I/O performance.***

- **•Reduced cabling*:In a virtualized I/O environment, only one cable is needed to connect servers to both storage and network traffic. This can reduce data center cabling.***

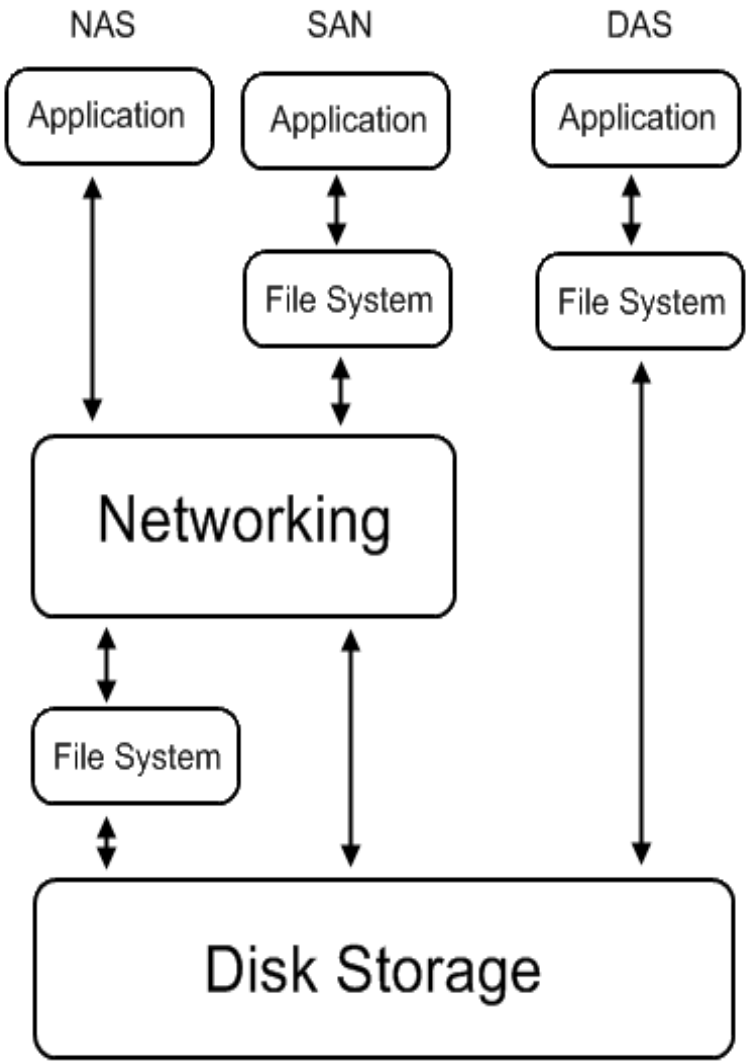
- **•Increased density*:I/O Virtualization increases the practical density of I/O by allowing more connections to exist within a given space.***

Storage Virtualization



Introduction

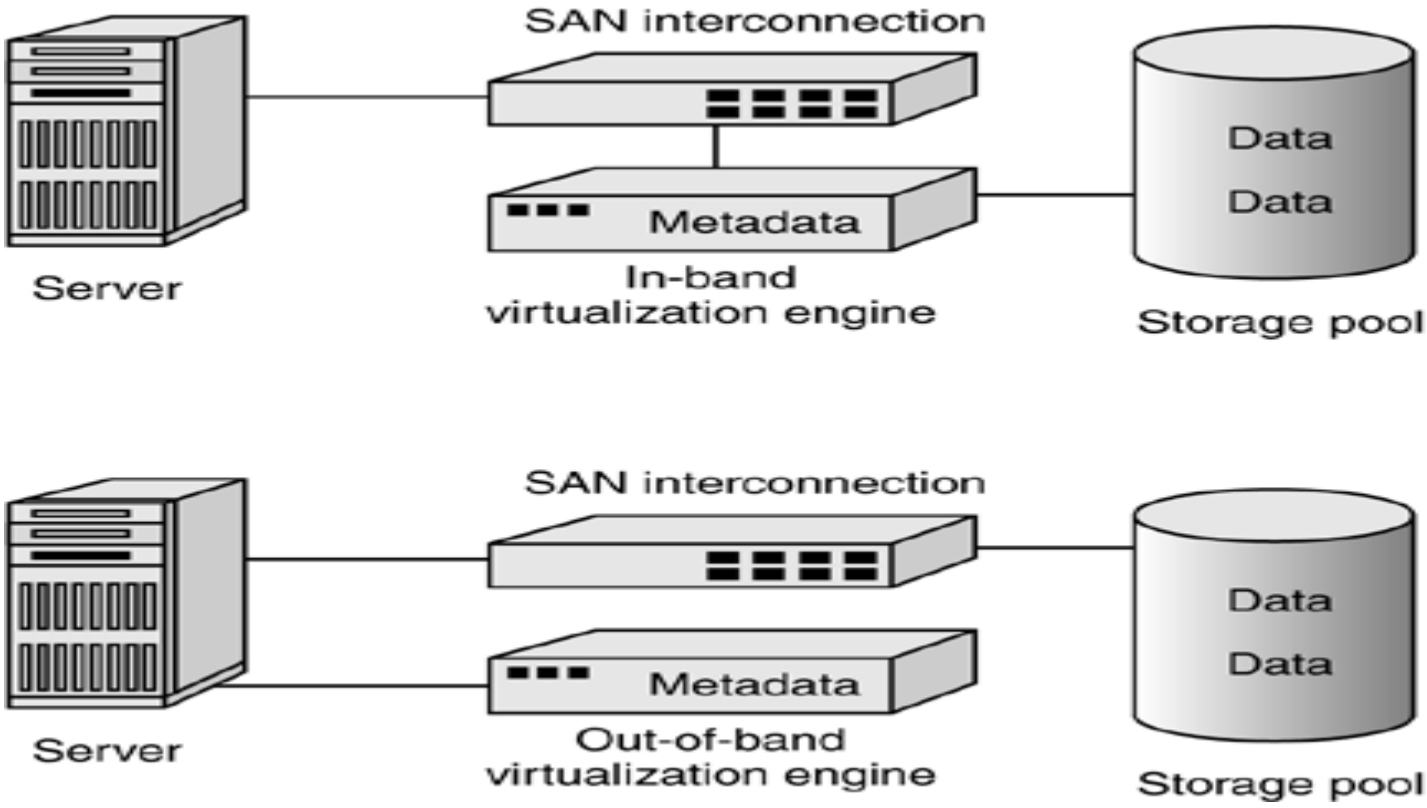
- Common storage architecture :
  - DAS - Direct Attached Storage
    - Storage device was directly attached to a server or workstation, without a storage network in between.
  - NAS - Network Attached Storage
    - File-level computer data storage connected to a computer network providing data access to heterogeneous clients.
  - SAN - Storage Area Network
    - Attach remote storage devices to servers in such a way that the devices appear as locally attached to the operating system.



Storage Virtualization

- What is created ?
  - Block Virtualization
  - Disk Virtualization
  - Tape, Tape Drive, Tape Library Virtualization
  - File System/ File/ Record Virtualization
  - Other device Virtualization
- Where it is done?
  - Host-based, Server based Virtualization
  - Network-based Virtualization
  - Storage Device, Storage subsystem Virtualization
- How it is implemented ?
  - In-band Virtualization
  - Out-of-band Virtualization

Storage Virtualization Implementation : In band / Out of band



Out Of Band Virtualization

- In an out-of-band implementation, the data flow is separated from the control flow. This is achieved by separating the data and meta-data (data about the data) into different places.
- Out-of-band virtualization involves moving all mapping and locking tables to a separate server (the meta-data controller) that contains the meta-data of the files.
- In an out-of-band solution the servers request authorization to data from the meta-data controller, which grants it, handles locking, and so on. Once they are authorized, servers access the data directly without any meta-data controller intervention. Once a client has obtained access to a file, all I/O will go directly over the SAN to the storage devices.
- For many operations, the meta-data controller does not even intervene. Separating the flow of control and data in this manner allows the I/O to use the full bandwidth that a SAN provides, while control could go over a separate network or routes in the SAN that are isolated for this purpose.
- Also known as ***asymmetric***, virtualization devices are sometimes called metadata servers.
- Require additional software in the host which knows the first request location of the actual data.

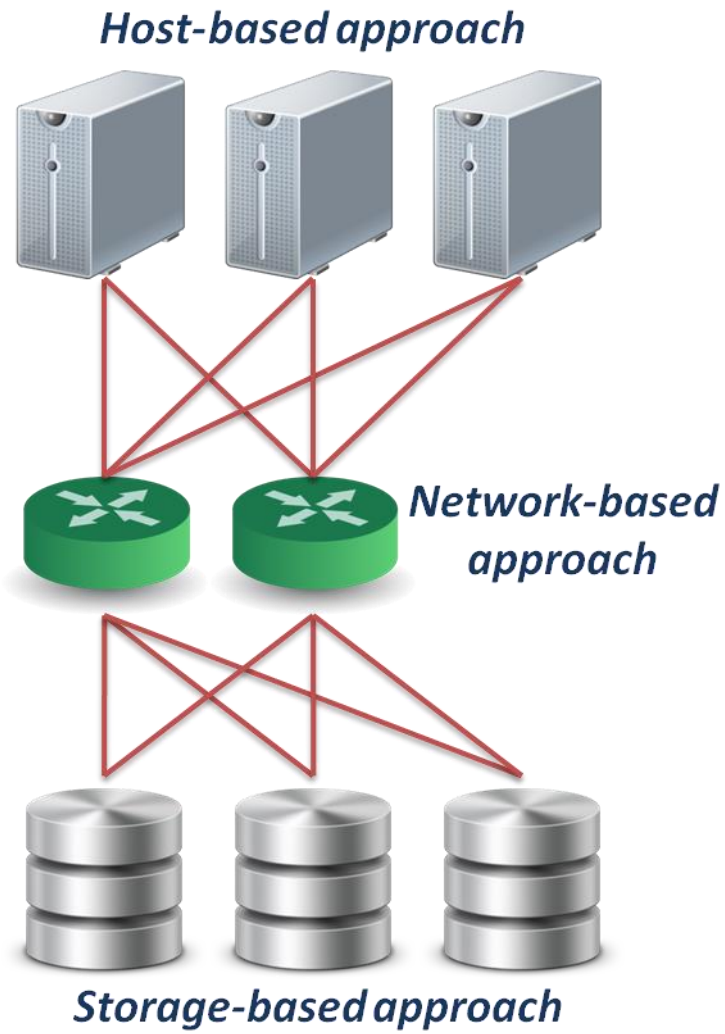
- Other advantages include:
  - Releasing the customer from a particular vendor's storage
  - Providing storage management for the SAN
  - Offering excellent scalability
  - Offloading host processing
  - Supporting storage management from multiple vendors
  - Integrating well with storage management software
  - Supporting multiple heterogeneous hosts
  - Relatively low overhead in the data path

In Band Virtualization

- When we implement an in-band virtual storage network, both data and control flow over the same path. Levels of abstraction exist in the data path, and storage can be pooled under the control of a domain manager.
- In general, in-band solutions are perceived to be simpler to implement, especially because they do not require special software to be installed in servers (other than conventional multi-pathing software).
- In-band solutions can also provide caching and advanced functions within the storage network. This can help to improve the performance of existing disk systems and can extend their useful life, and reduce the cost of new storage capacity by enabling the use of lower function, lower cost disk systems, without the loss of performance.
- Also known as ***symmetric***, virtualization devices actually sit in the data path between the host and storage.
- Hosts perform IO to the virtualized device and never interact with the actual storage device.
- Other advantages include:
  - Ability to offload function from the host
  - Providing storage management for the SAN
  - Performing performance optimizations in the data path
  - Supporting host systems not in a cluster
  - Supporting multiple heterogeneous hosts
  - Releasing the customer from a particular vendor's storage
  - Integrating with storage to create a better management picture
  - Offering excellent scalability

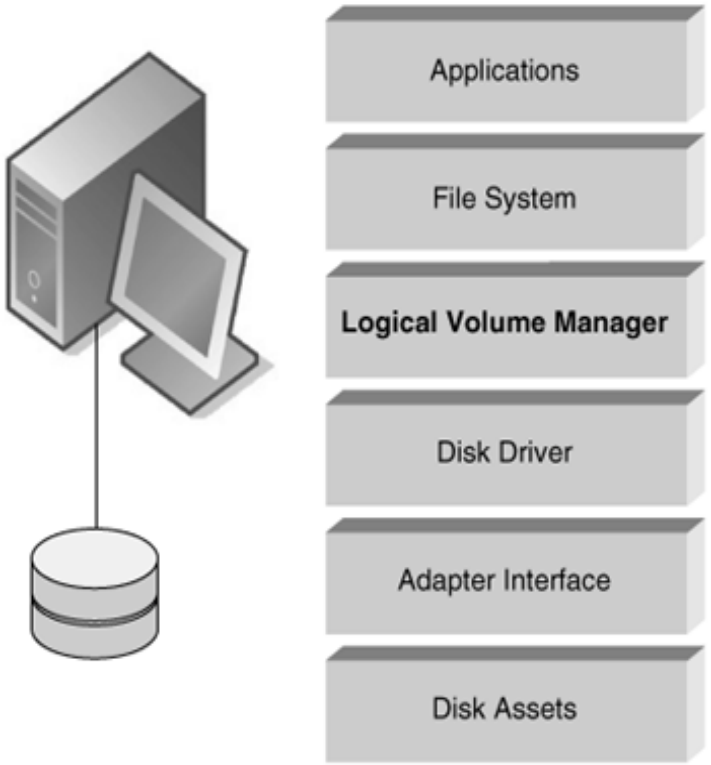
Where To Be Virtualized

- Different approaches :
  - Host-based approach
    - Implemented as a software running on host systems.
  - Network-based approach
    - Implemented on network devices.
  - Storage-based approach
    - Implemented on storage target subsystem.



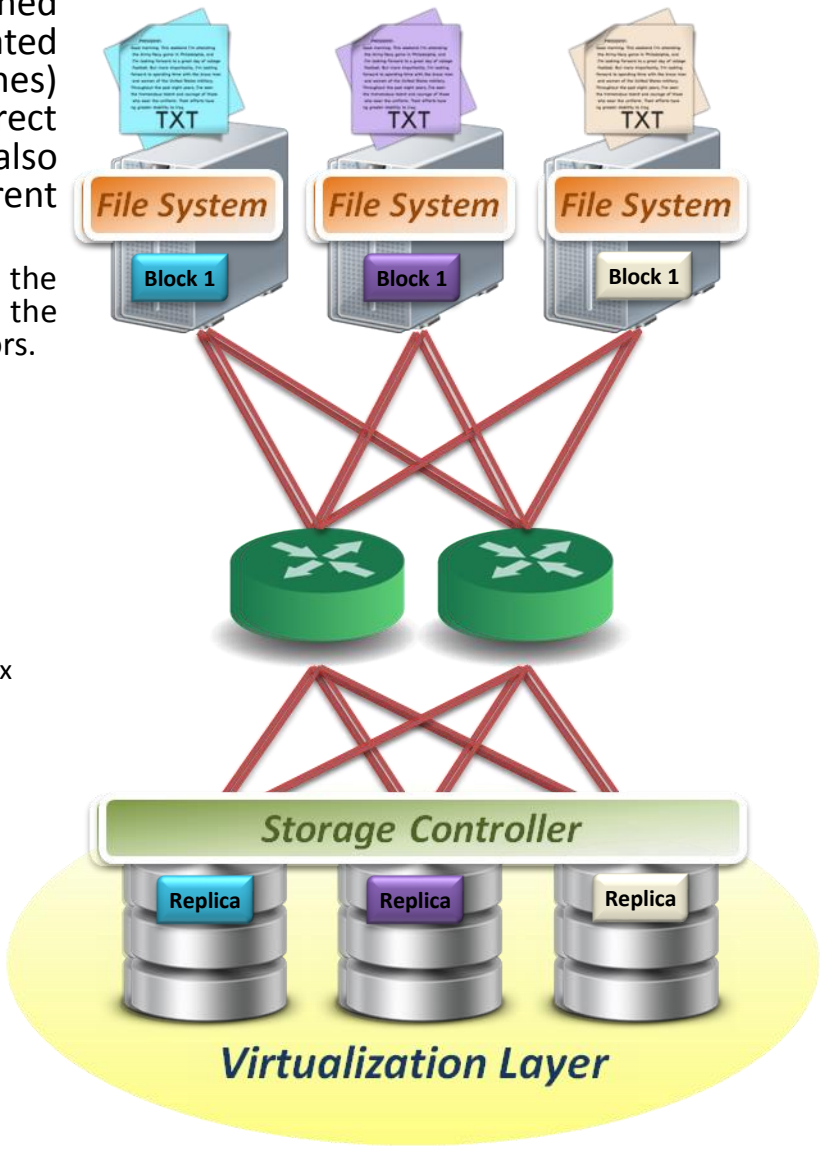
Storage Virtualization : Where it is done?

- Storage virtualization can be structured in three ways:
- **Host-based.** Here, physical drives are handled by a traditional device driver, while a software layer above the device driver intercepts I/O requests, looks up metadata and redirects I/O.
  - Host-based virtualization requires additional software running on the host, as a privileged task or process. In some cases volume management is built into the operating system, and in other instances it is offered as a separate product.
  - **Pros**
    - Simple to design and code
    - Supports any storage type
    - Improves storage utilization without thin provisioning restrictions
  - **Cons**
    - Storage utilization optimized only on a per host basis
    - Replication and data migration only possible locally to that host
    - Software is unique to each operating system
    - No easy way of keeping host instances in sync with other instances
    - Traditional Data Recovery following a server disk drive crash is impossible
  - Examples
    - LVM, NFS



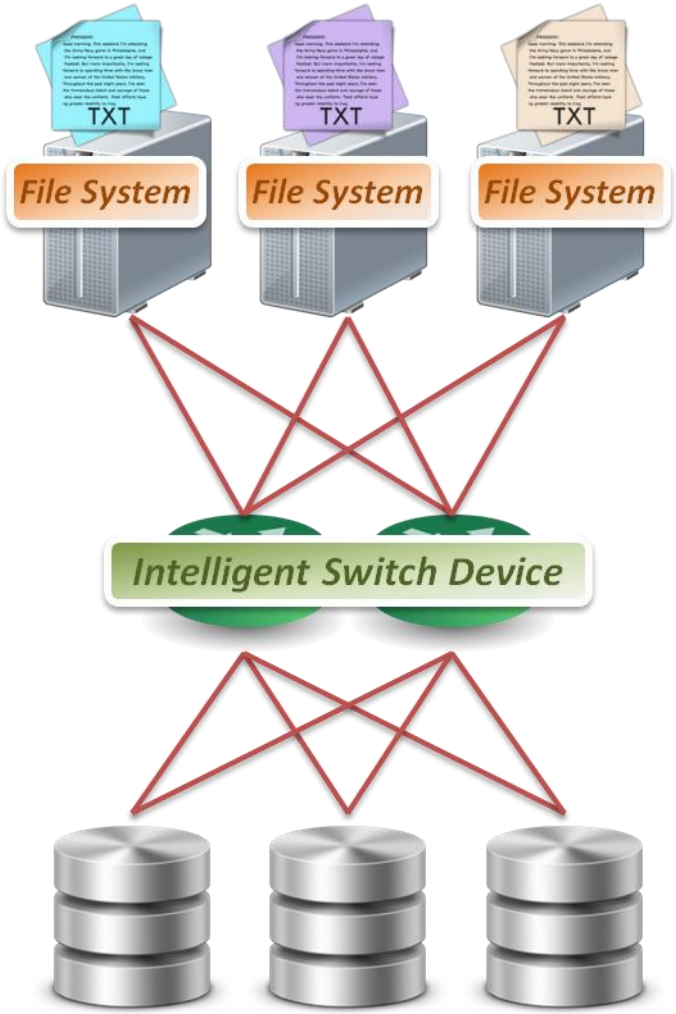
Where To Be Virtualized

- **Storage-device-based.** In this type of setup, virtualization can be built into the storage fabric; for example, newer RAID controllers allow other storage devices to be attached downstream. A primary storage controller (usually a dedicated hardware appliance, though some systems now use switches) handles pooling and manages metadata, allowing the direct attachment of other storage controllers. Such systems may also provide replication and migration services across different controllers.
  - A primary storage controller provides the services and allows the direct attachment of other storage controllers. Depending on the implementation these may be from the same or different vendors.
  - **Pros**
    - No additional hardware or infrastructure requirements
    - Provides most of the benefits of storage virtualization
    - Does not add latency to individual I/Os
  - **Cons**
    - Storage utilization optimized only across the connected controllers
    - Replication and data migration only possible across the connected controllers and same vendors device for long distance support
    - Downstream controller attachment limited to vendors support matrix
    - I/O Latency, non cache hits require the primary storage controller to issue a secondary downstream I/O request
    - Increase in storage infrastructure resource, the primary storage controller requires the same bandwidth as the secondary storage controllers to maintain the same throughput
  - Examples
    - Disk array products



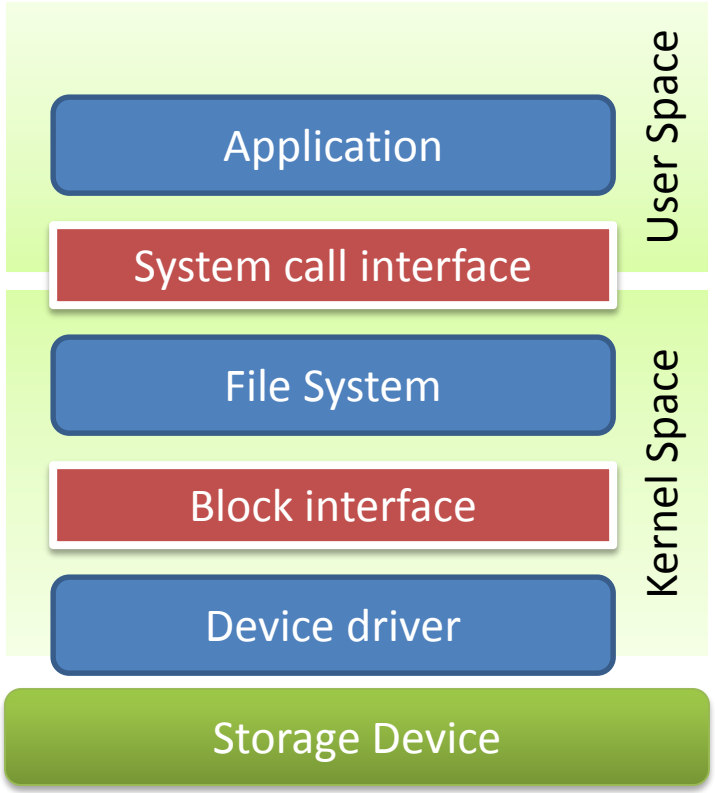
Storage Virtualization : Where it is done?

- **Network-based.** In this configuration, storage virtualization is viewed as a network-based device, generally using Fibre Channel networks connected as a SAN. Here, too, an appliance or switch-based implementation is most common.
  - The virtualization device sits in the SAN and provides the layer of abstraction between the hosts performing the I/O and the storage controllers providing the storage capacity.
  - **Pros**
    - True heterogeneous storage virtualization
    - Caching of data (performance benefit) is possible when in-band
    - Single management interface for all virtualized storage
    - Replication services across heterogeneous devices
  - **Cons**
    - Complex interoperability matrices - limited by vendors support
    - Difficult to implement fast meta-data updates in switched-based devices
    - Out-of-band requires specific host based software
    - In-band may add latency to I/O
    - In-band the most complicated to design and code
  - Examples
    - IBM SVC (SAN Volume Controller), EMC Invista



What To Be Virtualized

- Layers can be virtualized
  - File system
    - Provide compatible system call interface to user space applications.
  - Block device
    - Provide compatible block device interface to file system.
    - Through the interface such as SCSI, SAS, ATA, SATA, etc.

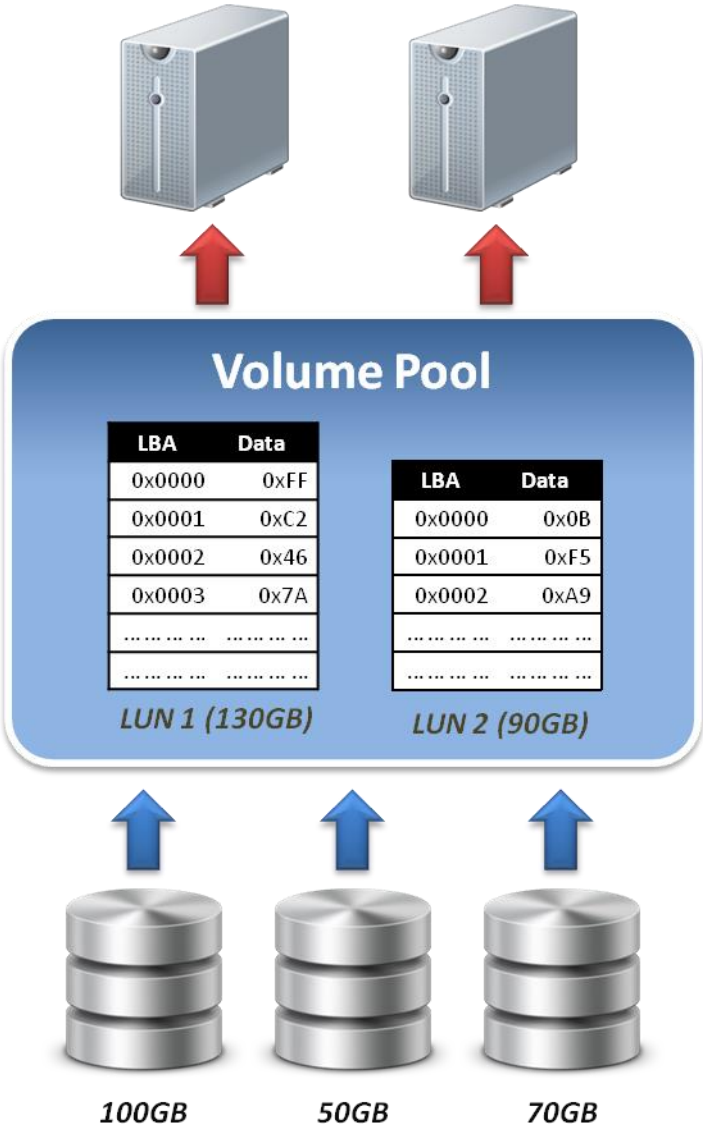


Storage Virtualization: How it is done?

- Within the context of a storage system, there are two primary types of virtualization that can occur:
- **Block virtualization** used in this context refers to the abstraction (separation) of logical storage (partition) from physical storage so that it may be accessed without regard to physical storage or heterogeneous structure. This separation allows the administrators of the storage system greater flexibility in how they manage storage for end users.
- **File virtualization** addresses the NAS challenges by eliminating the dependencies between the data accessed at the file level and the location where the files are physically stored. This provides opportunities to optimize storage use and server consolidation and to perform non-disruptive file migrations.

Block Device Level

- Data block level virtualization
  - LUN & LBA
    - A single block of information is addressed using a logical unit identifier (LUN) and an offset within that LUN, which known as a Logical Block Address (LBA).
  - Apply address space remapping
    - The address space mapping is between a logical disk and a logical unit presented by one or more storage controllers.



File System Level

- File system level virtualization
  - File system maintains metadata (*i*-node) of each file.
  - Translate file access requests to underlining file system.
  - Sometime divide large file into small sub-files (chunks) for parallel access, which improves the performance