

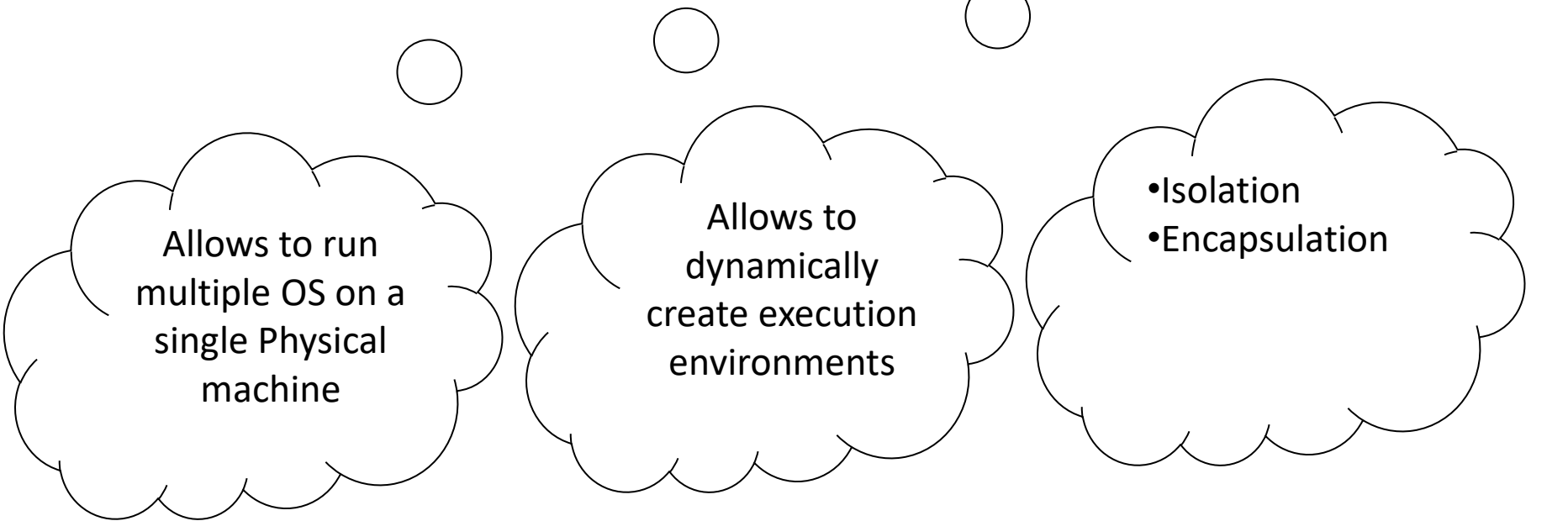
Module-1

Introduction

Virtualization Definition

What is Virtualization?

“ Virtualization is a framework or methodology of dividing the resources of a computer into multiple execution environments, by applying hardware and software partitioning, time-sharing, partial or complete machine simulation and/or emulation”

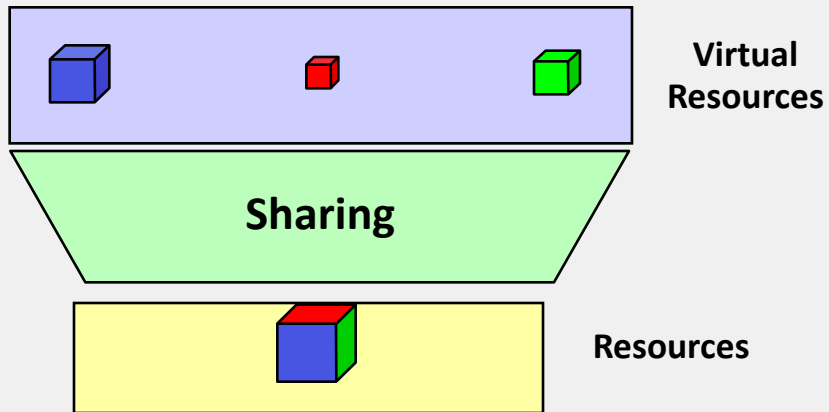


Allows to run
multiple OS on a
single Physical
machine

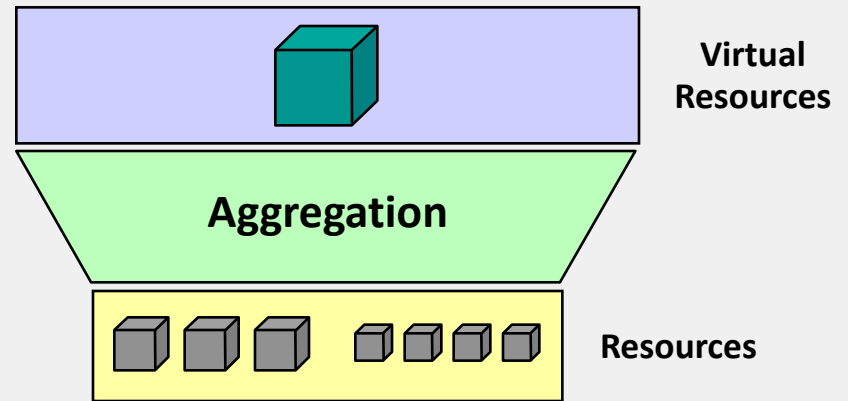
Allows to
dynamically
create execution
environments

- Isolation
- Encapsulation

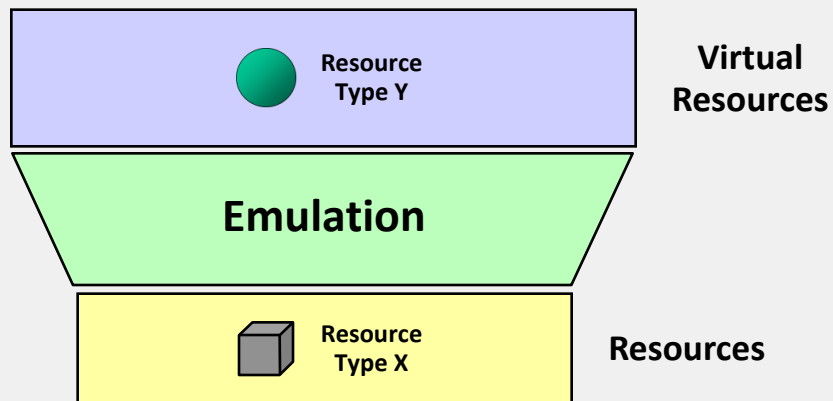
Virtualization Functions



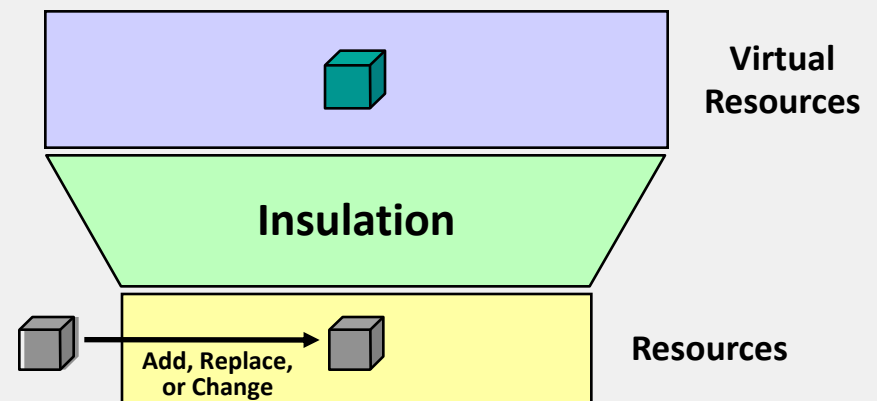
Examples: **LPARs, VMs, virtual disks, VLANs**
Benefits: **Resource utilization, workload manageability, flexibility, isolation**



Examples: **Virtual disks, IP routing to clones**
Benefits: **Management simplification, investment protection, scalability**



Examples: **Arch. emulators, iSCSI, virtual tape**
Benefits: **Compatibility, investment protection, interoperability, flexibility**



Examples: **Spare CPU subst., CUoD, SAN-VC**
Benefits: **Continuous availability, flexibility, investment protection**

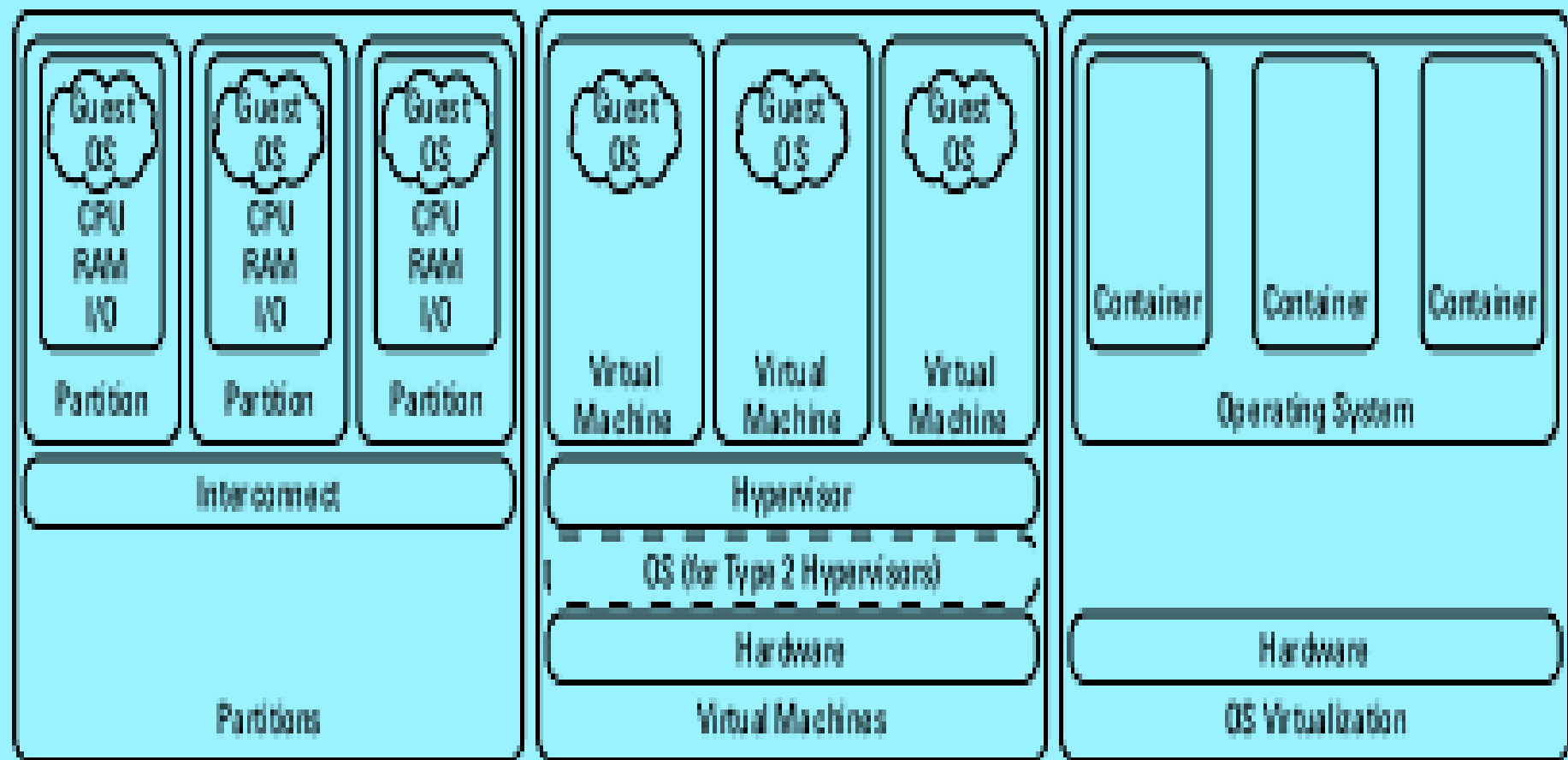
History of Virtualization

- In 1960, mainframe computers are used to run a single application at a time and waste a lot of resources.
- Split the resources of a mainframe computer and to execute multiple applications.
- In 1980, x86 becomes a dominant architecture, introduces master-slave paradigm and virtualization no longer needed.

Continued ...

- 1990, massive growth in computer technology and introduces a lot of challenges.
 - Low hardware infrastructure utilization;
 - Rising physical infrastructure costs;
 - Rising IT management costs;
 - Insufficient disaster protection; and
 - High maintenance and costly end-user desktops.

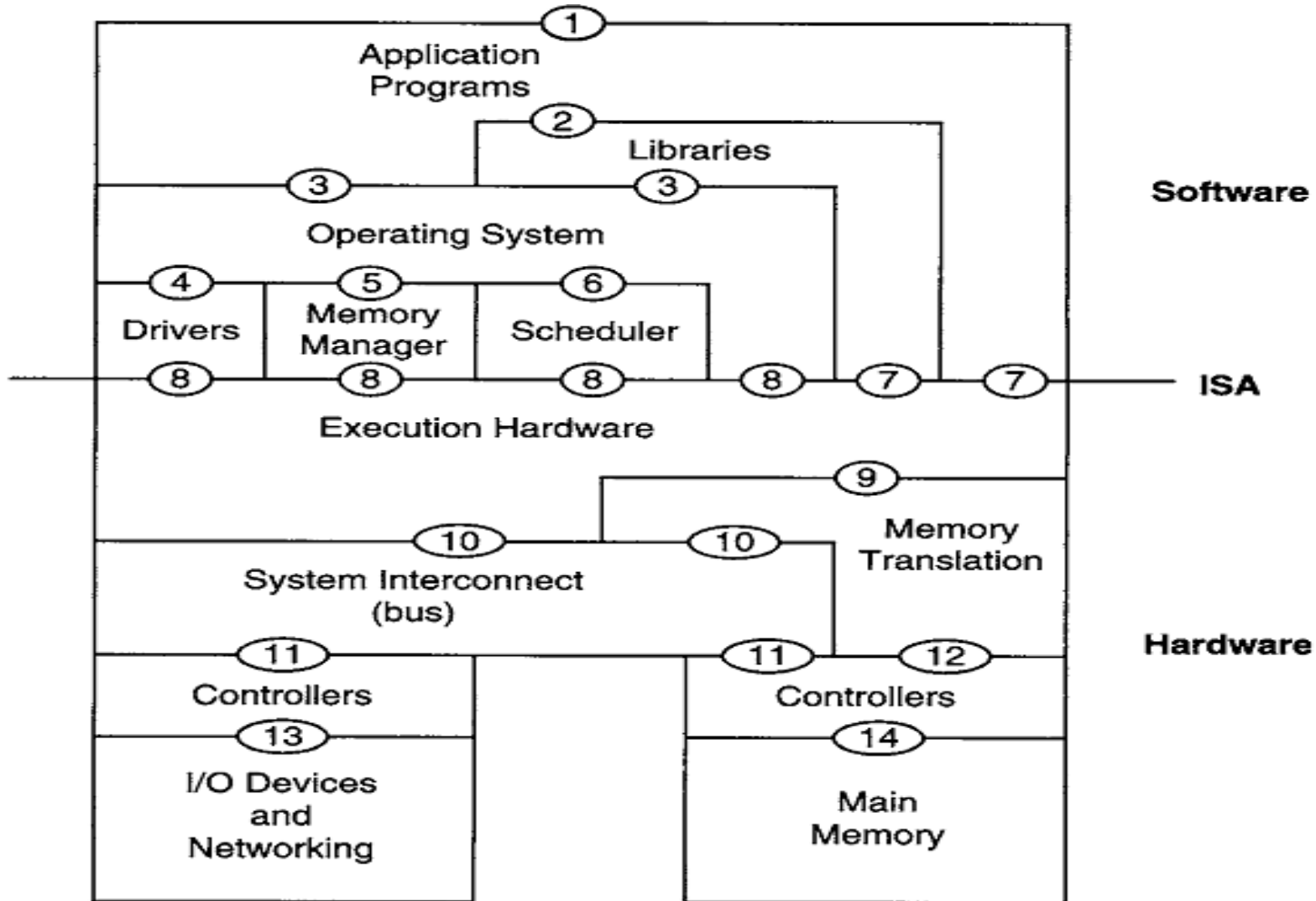
Traditional Vs Contemporary Virtualization process

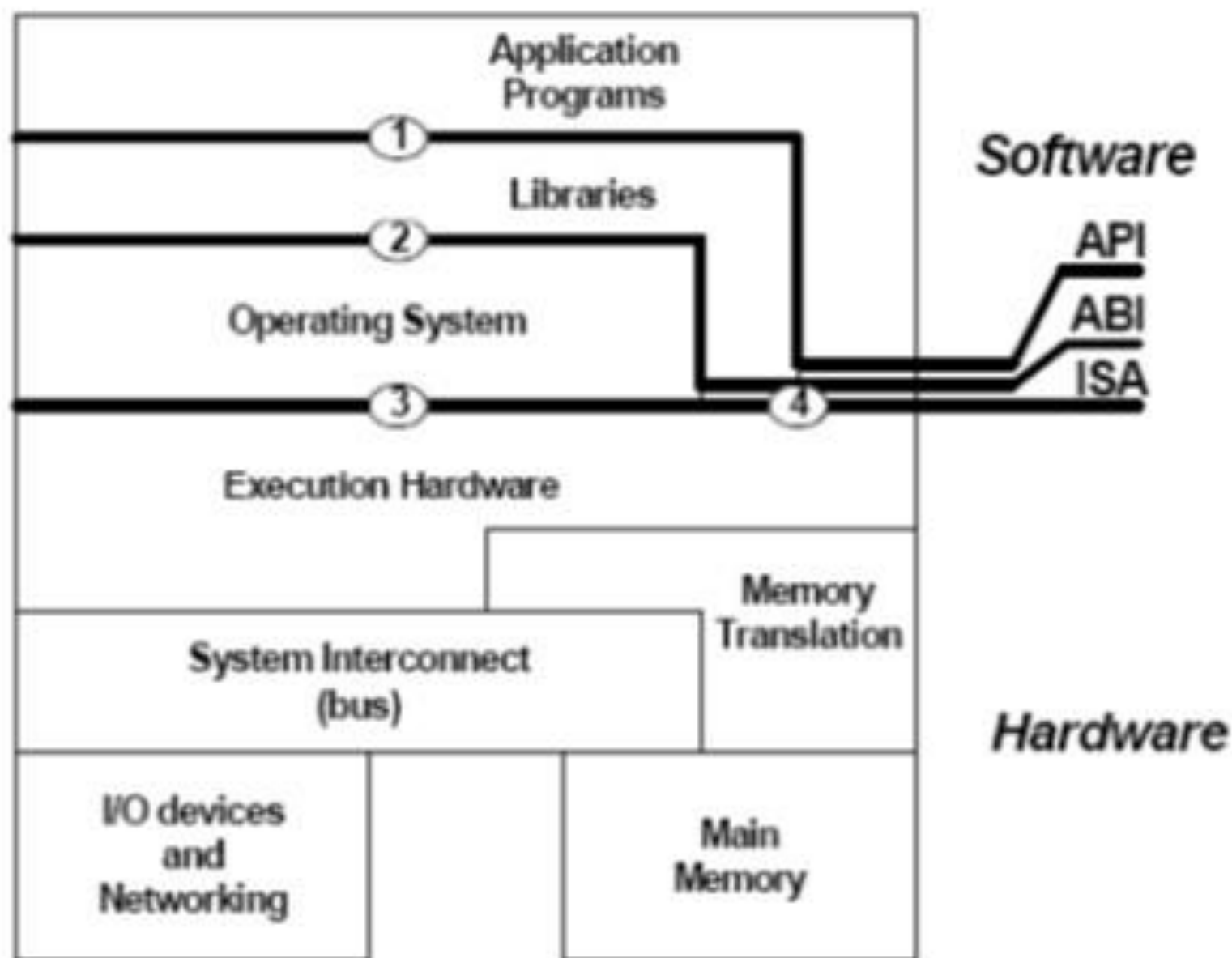


← More Isolation

More Flexibility →

Computer System Architecture

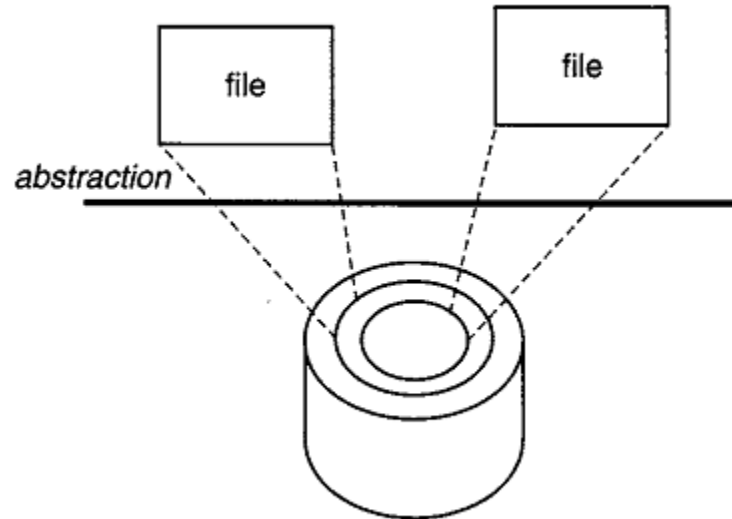




Abstraction and interfaces of Computer architecture

- The key to managing complexity in computer systems is their division into **levels of abstraction** separated by **well-defined interfaces**.
- Levels of abstraction allow implementation details at lower levels of a design to be ignored or simplified, thereby simplifying the design of components at higher levels.

Levels of abstraction



straction of a Disk. A level of abstraction provides a simplified interfa

Interfaces

- Well-defined interfaces allow computer design tasks to be decoupled so that teams of hardware and software designers can work more or less independently.
- EX: Hardware engineers focused to implement the IA32 instruction set on hardware, whereas the software engineers developed a compiler that convert HLL to the IA32 instruction set.

Drawbacks of Interfaces

- Subsystems and components designed to specifications for one interface will not work with those designed for another.
 - e.g., Intel IA-32 and IBM PowerPC
- Application programs, when distributed as program binaries, are tied to a specific instruction set and operating system.
- An operating system is tied to a computer that implements specific memory system and I/O system interfaces.

Continued ...

- The implicit assumption is that the hardware resources of a system are managed by a single operating system. This binds all hardware resources into a single entity under a single management regime.
- In turn, limits the flexibility of the system, not only in terms of available application software, but also in terms of security and failure isolation, especially when the system is shared by multiple users or groups of users.

How to Solve?

Virtualization

- Virtualization provides a way of relaxing the foregoing constraints and increasing flexibility. When a system (or subsystem), e.g., a processor, memory, or I/O device, is virtualized, its interface and all resources visible through the interface are mapped onto the interface and resources of a real system actually implementing it.

Common terminologies

- User ISA
 - ISA exposed for user applications
- System ISA
 - ISA exposed for supervisory system
- Application Binary Interface
 - User ISA and System Call
- Instruction Set Architecture separates hardware from rest
- Application Binary Interface separates processes from rest

SENSITIVE INSTRUCTION

Category of instructions

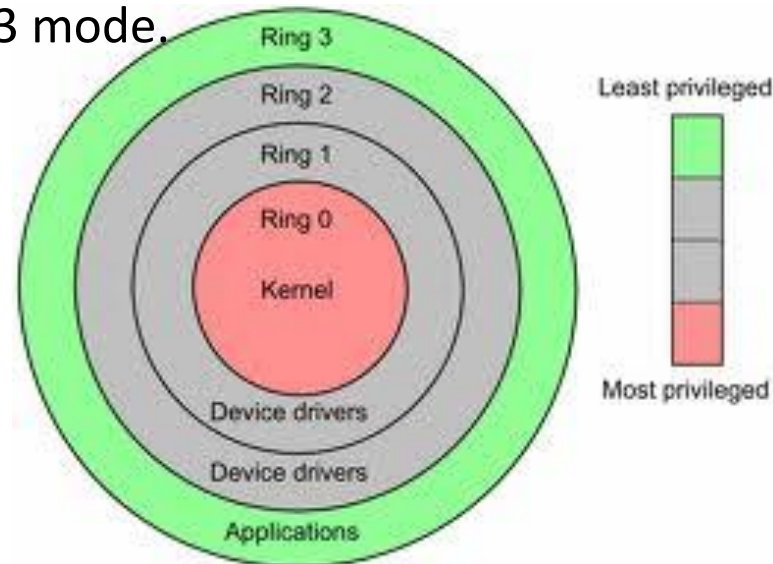
- In **architecture** field, the **CPU designers** separate instructions into different categories.
 - Privilege instruction
 - Those instructions are trapped if the machine is in user mode and are not trapped if the machine is in kernel mode.
 - ex: Instruction to modify page table base register
 - Non-Privilege instruction
 - All other instructions
 - ex: **Software interrupt**, Normal arithmetic operation
- In **virtualization** field, the **hypervisor designers** separate instructions into two categories.
 - Sensitive instruction
 - Those instructions that interact with hardware, which include control-sensitive and behavior-sensitive instructions.
 - ex: Instruction to modify page table base register, **Software Interrupt**, ...
 - Non-sensitive instruction
 - All other instructions
 - ex: Normal arithmetic operation, ...

Privilege instruction

- In modern computer architecture, CPU contains privilege instructions and non-privilege instructions.
- The OS designer use the **privilege instructions** and **non-privilege instructions** to separate between **kernel space which access hardware resource directly** and **user space which access hardware resource indirectly**.
- HOWEVER, which instructions are privilege is decided by CPU designers, OS designer cannot change that.
- If you execute privilege instruction in non-privilege mode, it will trigger an event and enter into the privilege mode.
 - This kind of behavior is also known as “trap”.

Privilege instruction

- Take x86 architecture for example:
 - Kernel mode (Ring 0)
 - CPU may perform any operation allowed by its architecture, including any instruction execution, IO operation, area of memory access, and so on.
 - Traditional OS kernel runs in Ring 0 mode.
 - User mode (Ring 1 ~ 3)
 - CPU can typically only execute a subset of those available instructions in kernel mode.
 - Traditional application runs in Ring 3 mode.



Privilege instruction

- Taking ARM architecture for another example:

- Kernel mode (Privilege Level 1, a.k.a. PL1)

- CPU may perform any operation allowed by its architecture, including any instruction execution, IO operation, area of memory access, and so on.
 - Traditional OS kernel runs in PL1.

- User mode (PL0)

- CPU can typically only execute a subset available instructions in kernel mode.
 - Traditional application runs in PL0 mode

User Space

Kernel Space

Secure PL0 User mode

Secure PL1 System mode Supervisor mode FIQ mode IRQ mode Undef mode Abort mode
--

Sensitive Instruction

- Those instructions that interact with hardware, which include control-sensitive and behavior-sensitive instructions
- Control sensitive instructions
 - Those that attempt to change the configuration of resources in the system.
- Behavior sensitive instructions
 - Those whose behavior or result depends on the configuration of resources (the content of the relocation register or the processor's mode).

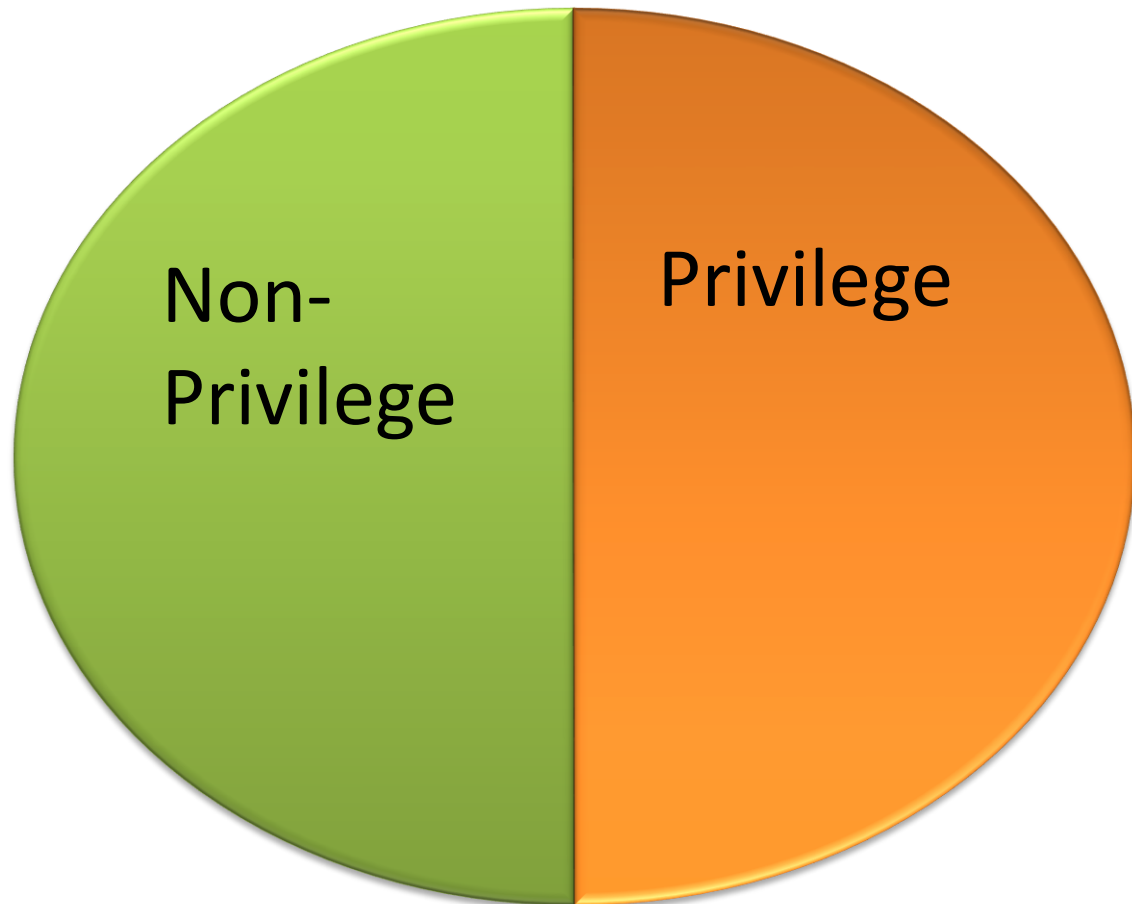
Example: ARMv6 ISA

- 1.Branch instructions
- 2.Data-processing instructions
- 3.Multiply instructions
- 4.Parallel addition and subtraction instructions
- 5.Extend instructions
- 6.Miscellaneous arithmetic instructions
- 7.Other miscellaneous instructions
- 8.Status register access instructions
- 9.Load and store instructions
- 10.Load and Store Multiple instructions
- 11.Semaphore instructions
- 12.Exception-generating instructions
- 13.Coprocessor instructions

Virtualizable CPU

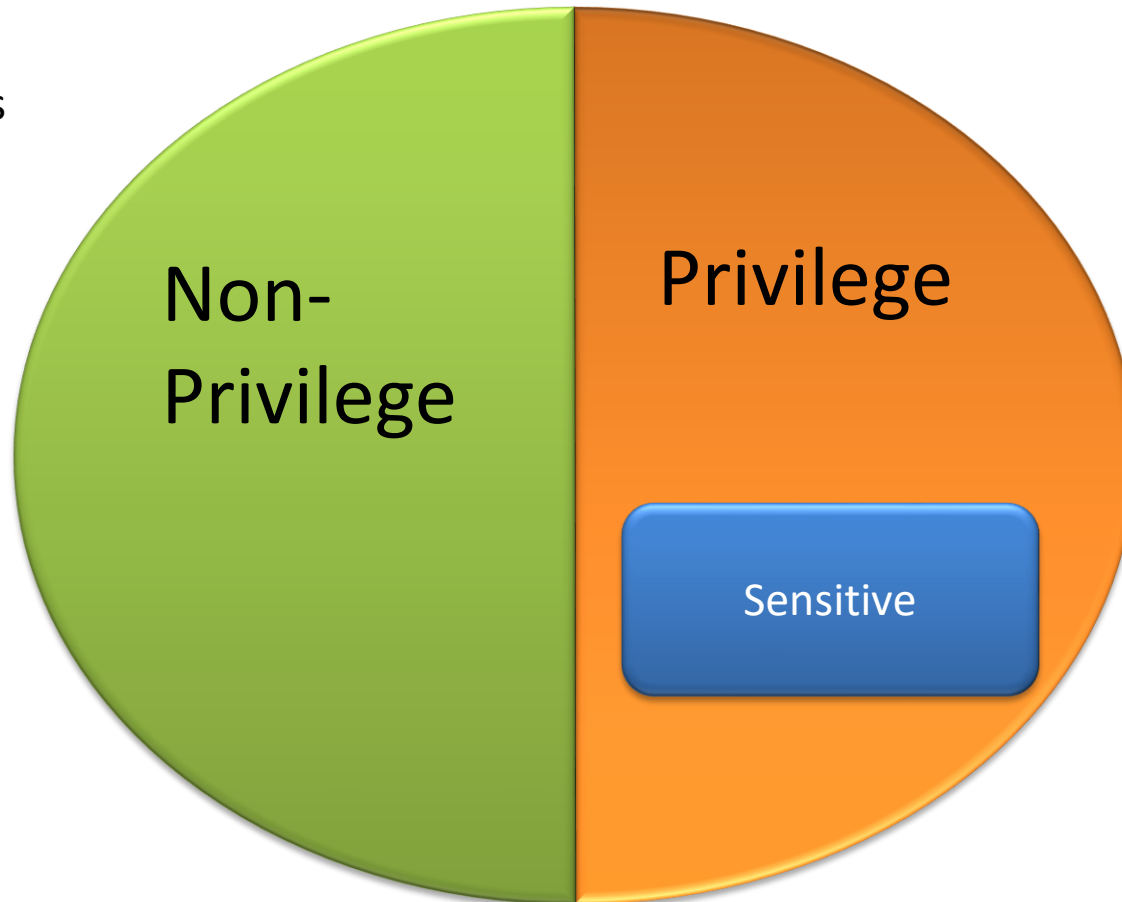
Privilege and Non-Privilege

ps: whole circle is a
set of all instructions



Privilege and Sensitive

ps: whole circle is a
set of all instructions



All sensitive are privilege: **Virtualizable CPU**

Virtualizable CPU

- All of sensitive instructions are privilege instructions.
- We call this kind of CPU as “Virtualizable CPU”
- For “Virtualizable CPU”, it is quite easy to implement hypervisor. All you have to do is to put hypervisor in privilege mode and Guest OS in non-privilege mode.
- When Guest OS wants to execute sensitive instructions, the execution will be trapped to hypervisor which is running on privilege mode automatically.
 - By this way, we can make sure that there is no chance for Guest OS to change any important hardware resource directly. All important hardware resource is under management by hypervisor.

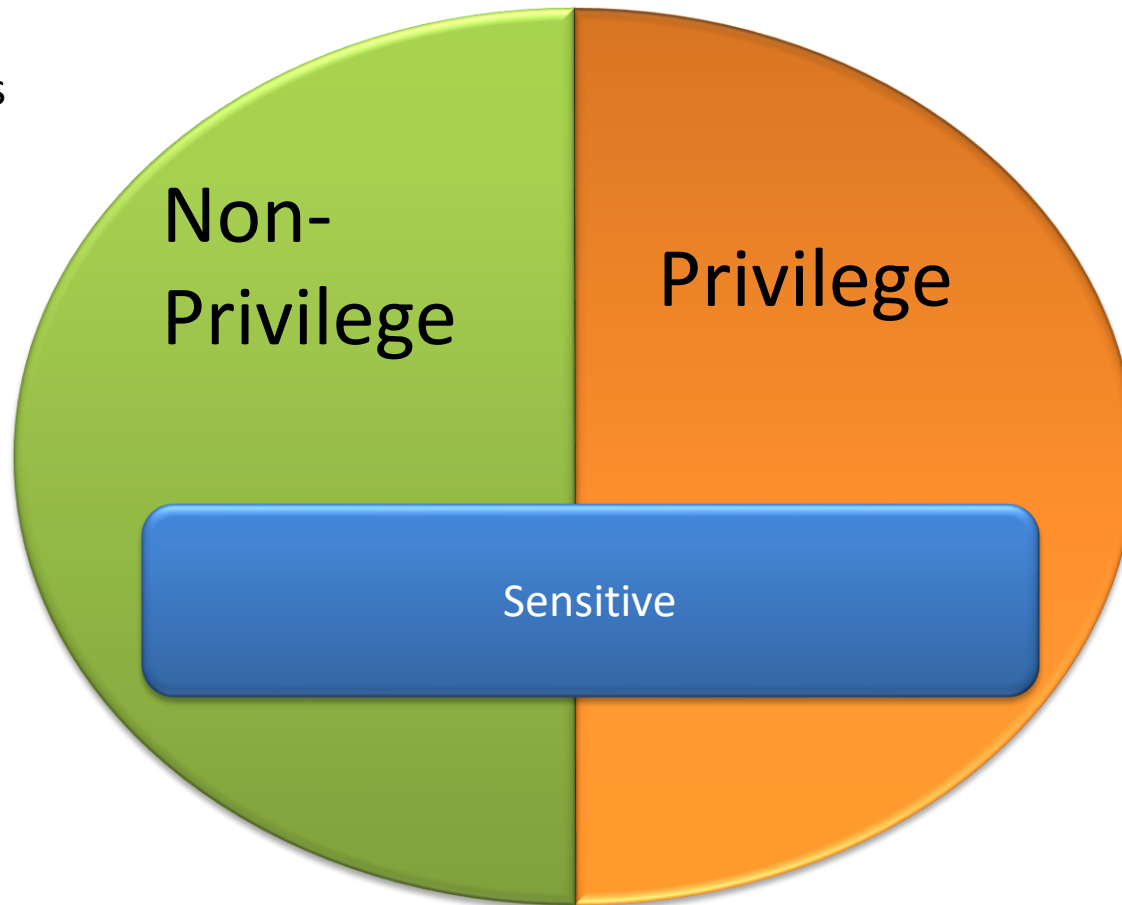
Virtualizable CPU

- Example:
 - IBM PowerPC
 - IBM S/390
 - CPU for IBM mainframe

Non-Virtualizable CPU

Privilege and Sensitive

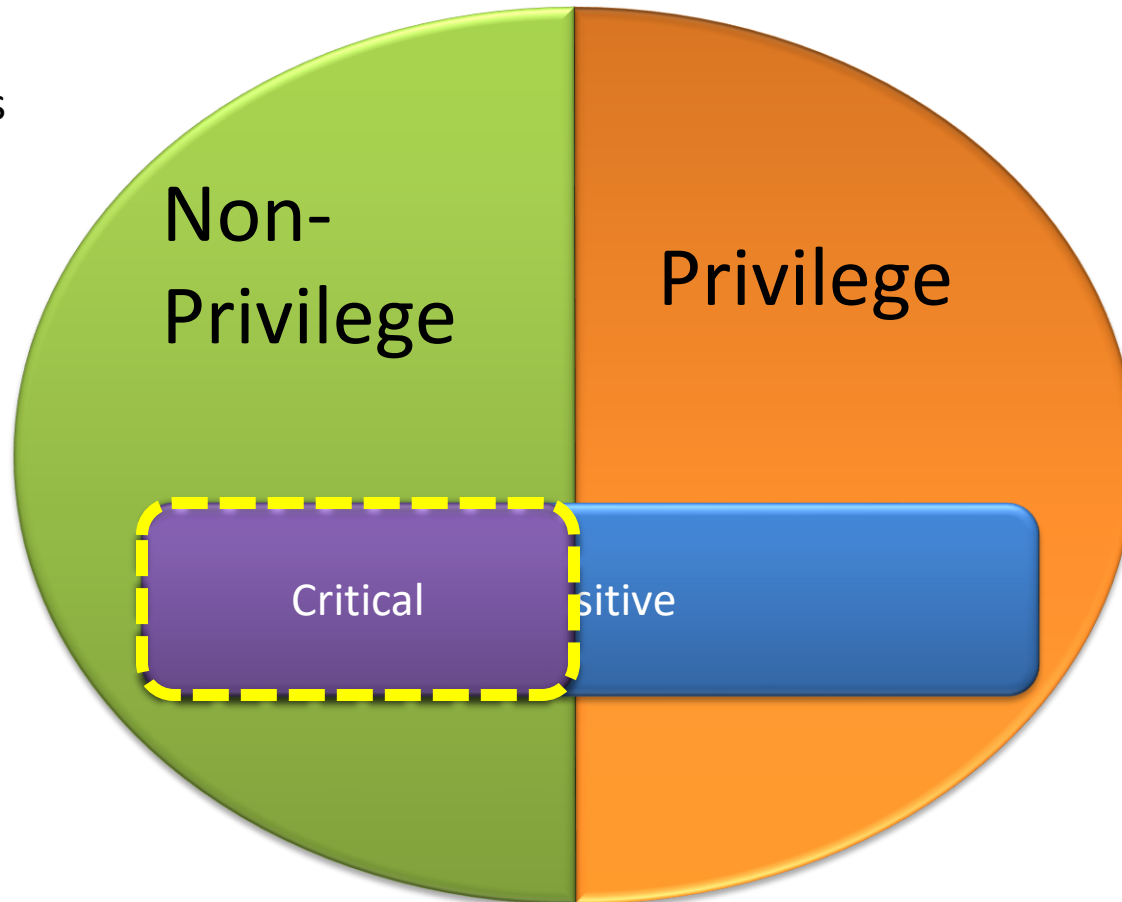
ps: whole circle is a
set of all instructions



Some sensitive are non-privilege: **Non-Virtualizable CPU**

Privilege and Sensitive

ps: whole circle is a
set of all instructions



Sensitive but Non-Privilege instruction is the problem.

We call "Sensitive but Non-Privilege instruction" as "Critical Instruction".

Non-Virtualizable CPU

- Some of sensitive instructions are privilege instructions. But some of sensitive instructions are non-privilege instructions.
- We call this kind of CPU as “Non-Virtualizable CPU”
- For “Non-Virtualizable CPU”, it is HARD to implement hypervisor.
- If you put hypervisor in privilege mode and Guest OS in non-privilege mode, when Guest OS wants to execute sensitive instructions,
 - For those privilege and sensitive instructions, they (which is) will be trapped into hypervisor which is running on privilege mode automatically.
 - For those critical instructions, they will NOT be trapped into hypervisor automatically.
- Example:
 - x86
 - ARM

Critical instruction annoy us!

- Critical instruction can be executed in privilege mode and non-privilege mode.
- The behaviors of critical instructions in privilege mode and non-privilege mode are different. It will cause problems.
- As a result, hypervisor designers have to let critical instructions be trapped to hypervisor, and let hypervisor emulate their behaviors.

Challenges in virtualizing x86 architecture

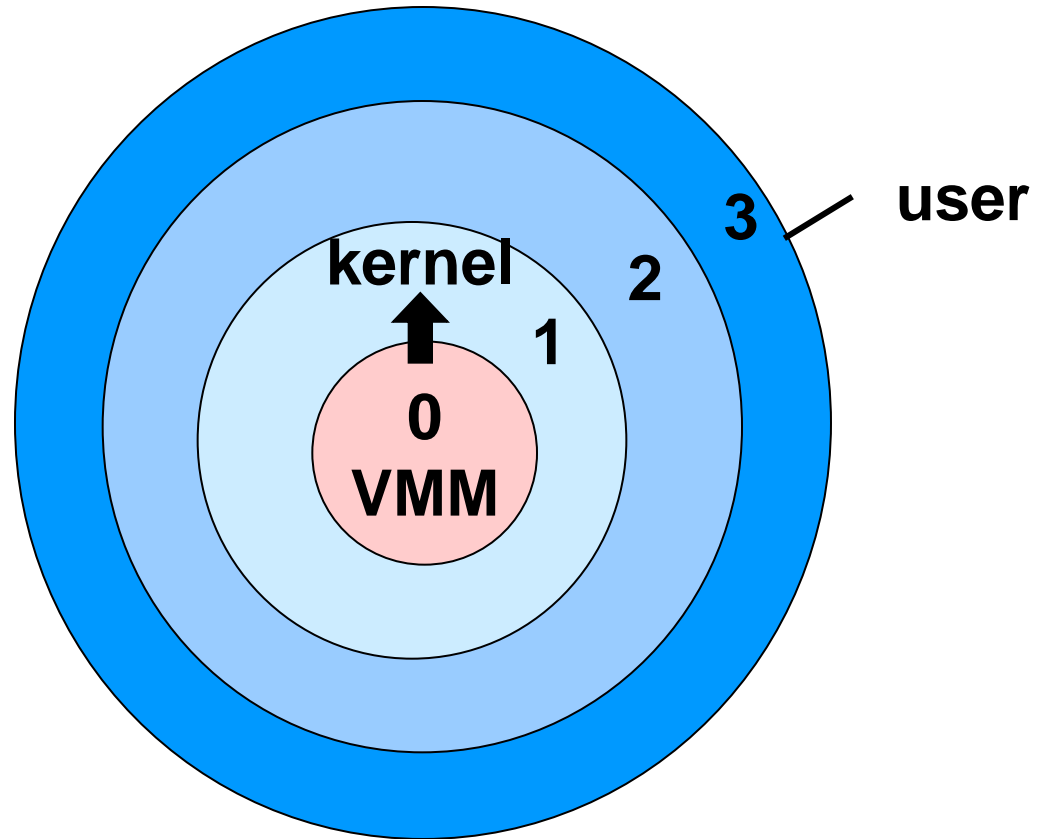
- Ring Aliasing
- Address space compression
- Non faulting access to privileged state (sensitive instructions)
- Interrupt virtualization

Virtualization through Ring Compression

Virtual Machine Monitor (VMM) runs at ring 0

Kernel(s) run at ring 1

Requires that CPU is virtualizable

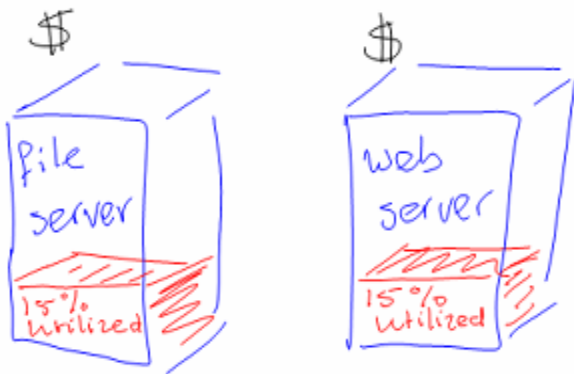


Benefits of Virtualization

- Easy manageability
- Elimination of Compatibility issues
- Fault isolation
- Improved security
- Improved utilization
- Portability
- Testing
- Rapid deployment
- Reduced Costs

Continued ...

Before virtualization:

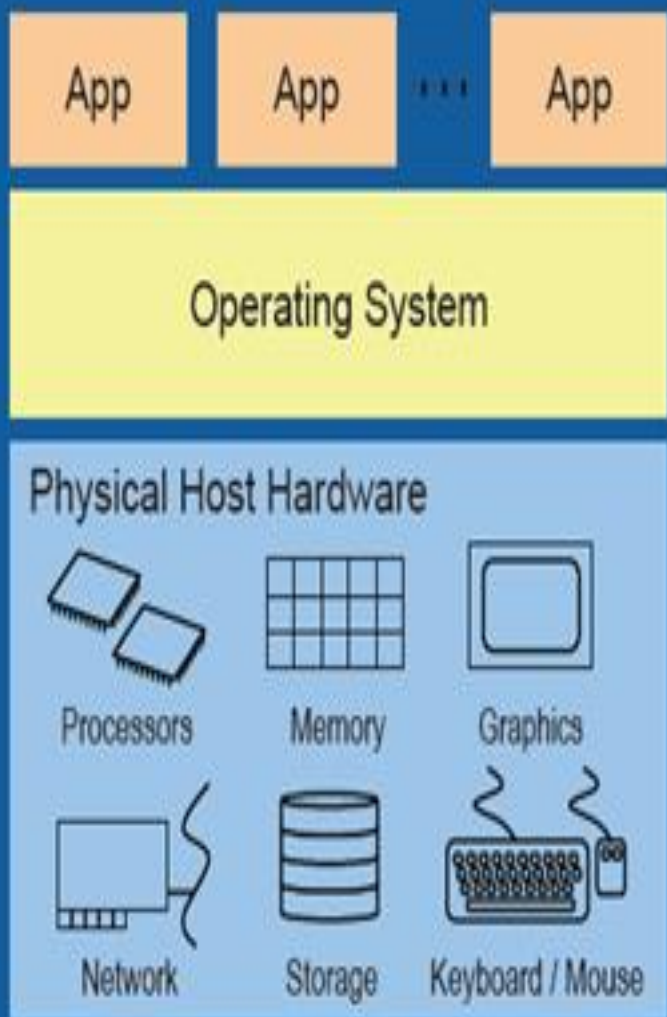


2x purchase
2x maintenance
2x depreciation
2x floor space
2x energy usage

After virtualization:

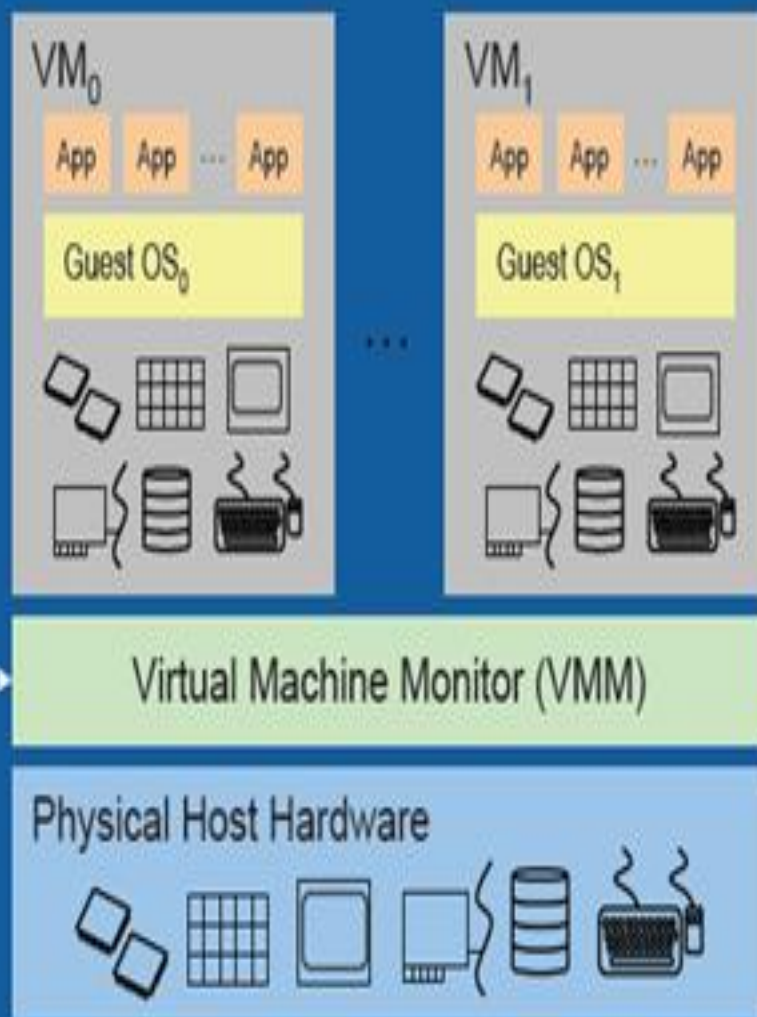


1 x purchase
1 x maintenance
1 x depreciation
1 x floor space
1 x energy usage
⇒ better utilization



Without VMs: Single OS owns all hardware resources

A new layer of software...



With VMs: Multiple OSes share hardware resources

- Move to Virtualization

- To increase the resource utilization or Reasons for resource under utilization

- Windows NT was a monolithic operating system. Since many operations, especially application operations, were performed at the kernel level of the OS, administrators quickly discovered that when an application froze, it would often freeze the entire OS and cause a Blue Screen of Death (BSOD) for the server as a whole, This leads to an important decision, “One server, One application”.

- Even after Microsoft solved the issue, end-user customers refuse to have their application hosted on the same server as another, stating that they cannot trust the stability of the other application, or even that they do not want to lose service because another application will have a different update schedule than theirs

- Use server virtualization to reduce the physical footprint of their datacenter

Need for Virtualization

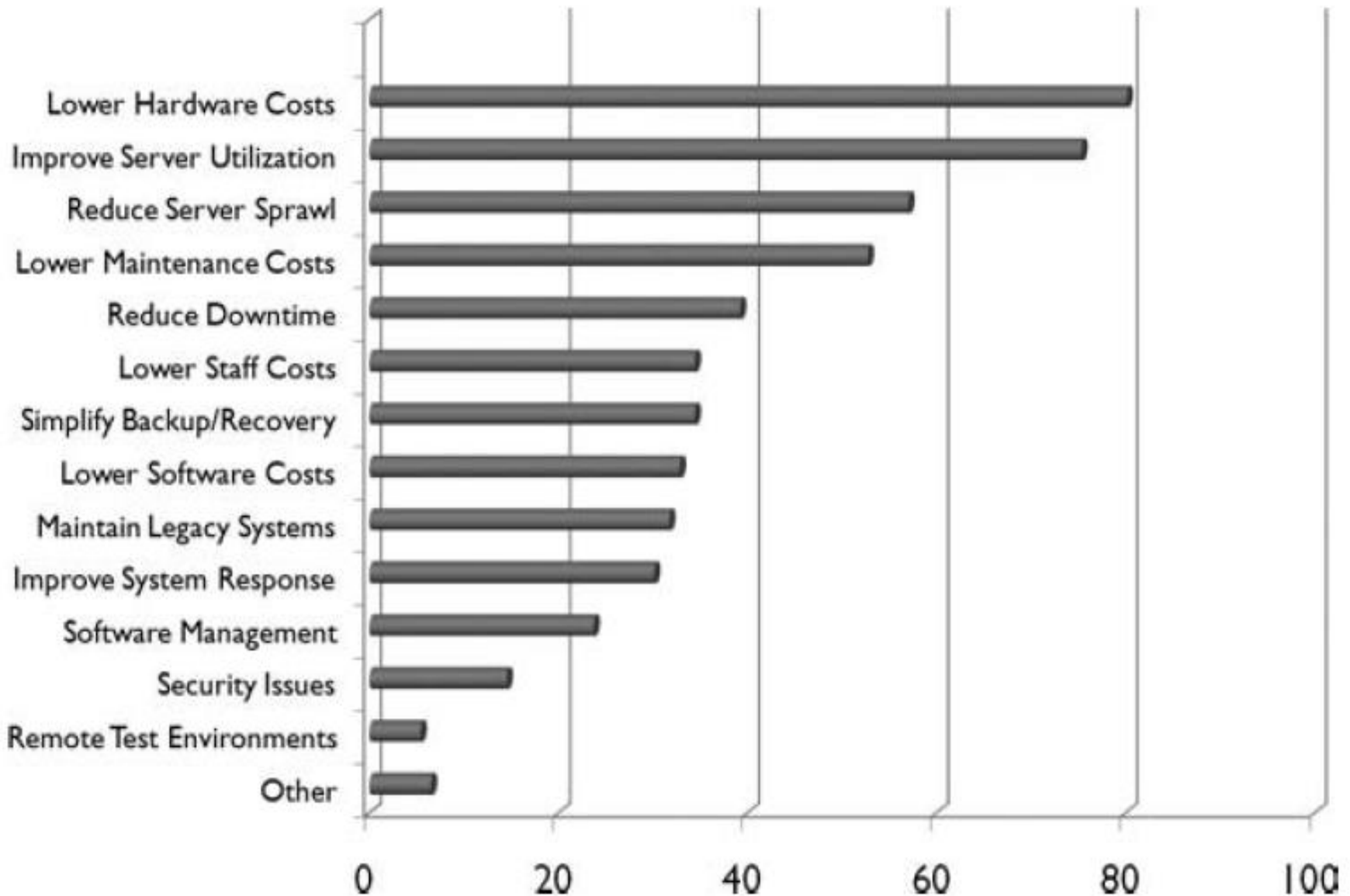
- More physical resources demands more power input as well as more cooling
- Physical servers are hard to provision
 - Need to rely on disk imaging to Windows Deployment Services (WDS) for faster provisioning
 - Ends up with 15% of utilization
- Disaster recovery to ensure business continuity is costly
 - Data replication to a server in a remote site.
- PCs are distributed throughout the entire organization, which makes them even harder to manage and repair
- Application conflicts after updating the OS.

Planning for Virtualization

- **When you plan for virtualization, you should answer the following questions**
 - What is virtualization?
 - Why would we need it?
 - How can it improve my business?
 - What types of virtualization technologies exist?
 - Which terms should I be familiar with?
 - What is the cost/benefit ratio of virtualization?
 - What new challenges will it bring to the datacenter?
 - How should I structure my virtualization solution?
 - Which applications or services are good virtualization candidates?
 - Which server platforms or form factors are best suited to support virtualization?
 - Am I missing anything else?

Virtualization Benefits

Common virtualization drivers



- **What do you mean by power rebate programs?**

- Power and utility companies also see the benefit from machine virtualization, since their customers can greatly reduce their power and cooling consumption ratios, ensuring that current power generation facilities can last longer and address the needs of a larger community of users
- Power and utility companies organize rebate programs for organizations implementing virtualization projects and removing physical servers from their datacenters.
 - Pacific Gas & Electric (PG&E)

- **Buying Green Technologies**

- Electronic Product Environmental Assessment Tool
(www.EPEAT.net)
 - Products fit into each category based on compliance with the following criteria:
 - Bronze products meet all of the required criteria.
 - Silver meets all of the required criteria and meets 50 percent of the optional criteria.
 - Gold meets all of the required criteria and all of the optional criteria.

- **Green Asset Disposal**

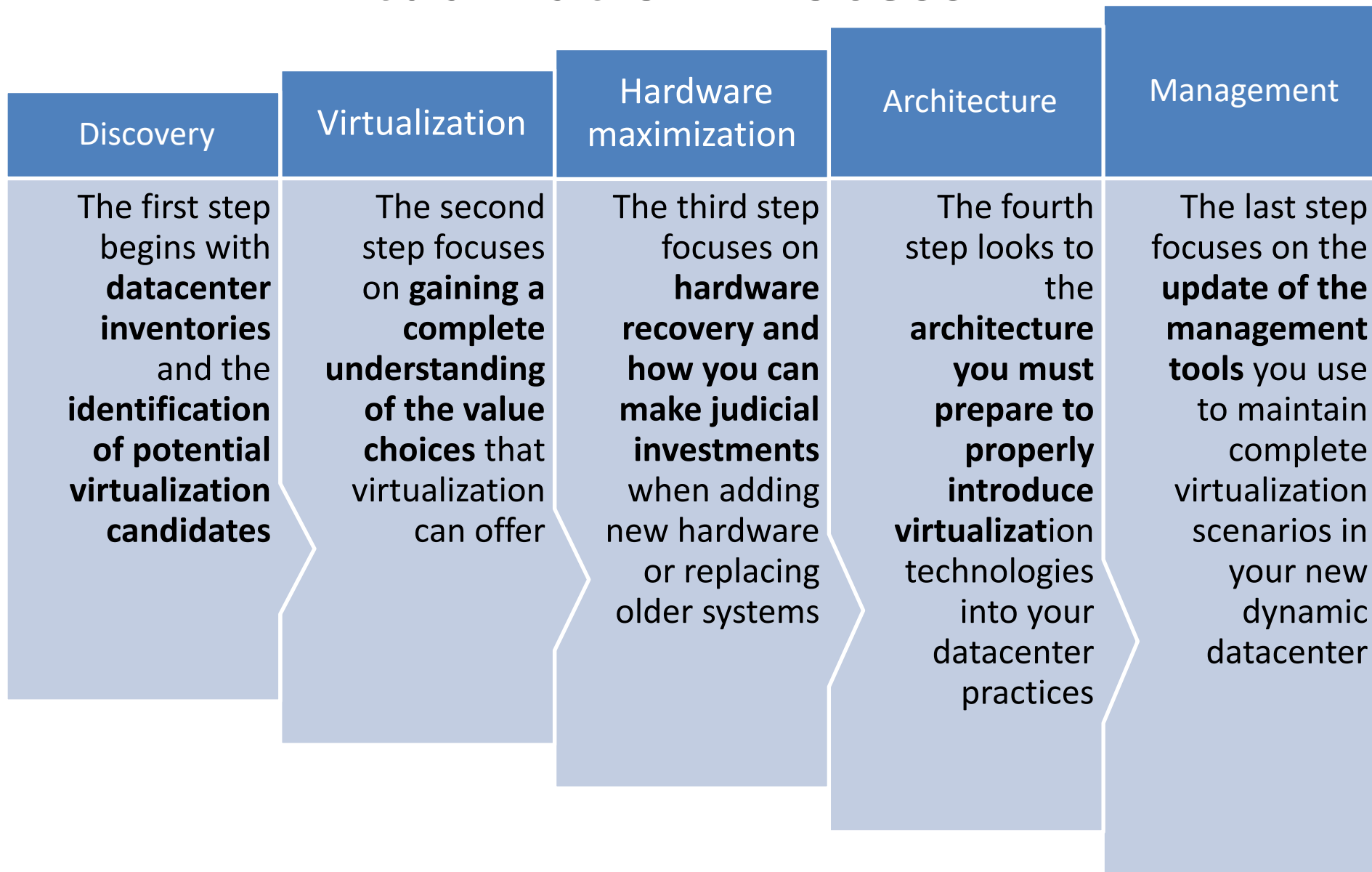
- if the purpose of your virtualization project is to reduce your carbon footprint, it doesn't make sense for you to reuse all of the equipment you are replacing.
- **Objective:** secure asset disposal while reducing the carbon footprint of the disposal process itself
 - secure information protection plans for obsolete hardware
 - Else, the information from the obsolete h/w can be easily accessed.
- **How to achieve?**
 - Green Asset Disposal (www.greenassetdisposal.com)
 - recycle, remarket, and even purchase the assets you need to pass on
 - Give information on how to properly dispose of technologies ranging from servers, PCs, and other obsolete IT equipment

- **Before you begin the virtualization project – Power consumption point of view**
 - Perform a power consumption analysis before the virtualization project begins.
 - This will identify how power is being used in the datacenter through traditional consumption methods.
 - Assess the potential power consumption reductions through virtualization
 - Transform physical servers into virtual machines, reducing the power consumption levels.
 - Perform post-project assessment
 - Compare the post-project assessment to the initial baseline to identify the new power consumption levels are identified

FIVE-STEP PROCESS

- **To get more return on your virtualization project, you need to know**
 - how different virtualization solutions fit together,
 - what each vendor offers,
 - what you actually need to implement to make the most of virtualization technologies in your datacenter while addressing your business needs
- The move to virtualization relies on five key steps:
 1. **Discovery** The first step begins with datacenter inventories and the identification of potential virtualization candidates.
 2. **Virtualization** The second step focuses on gaining a complete understanding of the value choices that virtualization can offer.
 3. **Hardware maximization** The third step focuses on hardware recovery and how you can make judicial investments when adding new hardware or replacing older systems.
 4. **Architecture** The fourth step looks to the architecture you must prepare to properly introduce virtualization technologies into your datacenter practices.
 5. **Management** The last step focuses on the update of the management tools you use to maintain complete virtualization scenarios in your new dynamic datacenter. Each of these building blocks brings you

Virtualization Process



Step 1 - Discovery



```
graph TD; A[Inventory] --> B[Scan for Potential Virtualization Candidates]; B --> C[Categorize Server Resources]; C --> D[Rationalize Everything];
```

Inventory

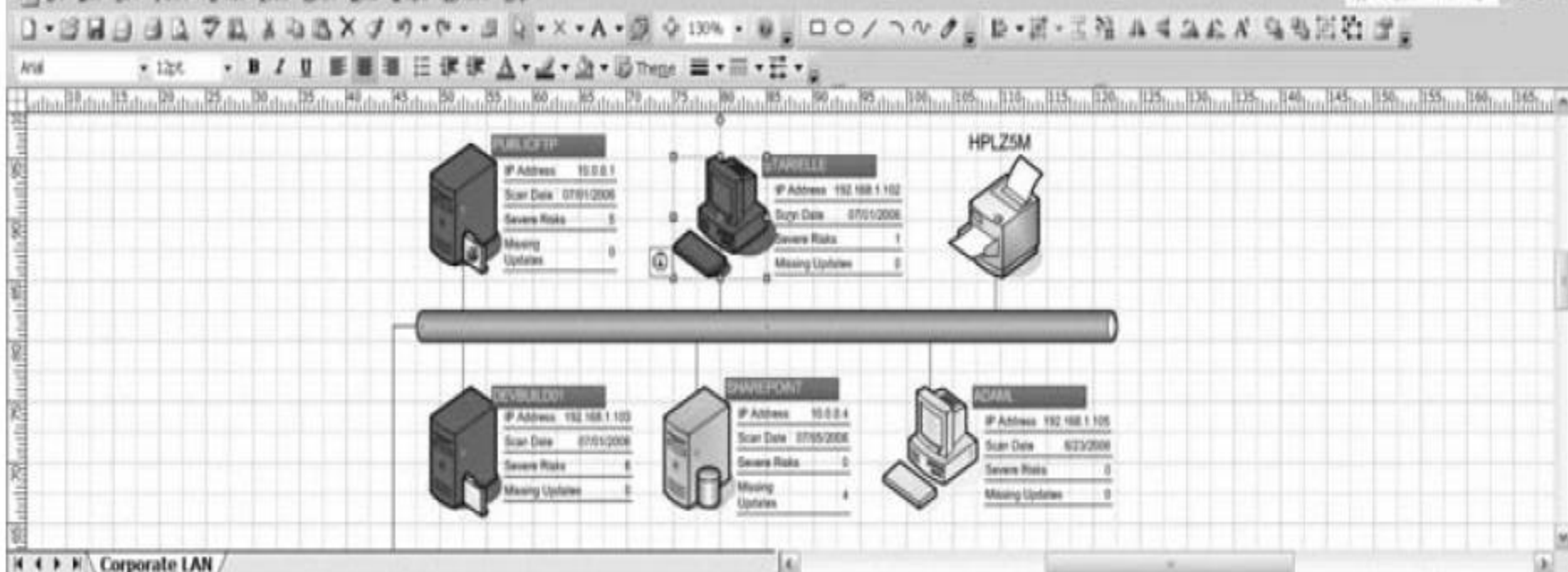
Scan for Potential Virtualization Candidates

Categorize Server Resources

Rationalize Everything

Step 1 : Discovery Inventory

- **Importance of Inventory**
 - How can you manage what you don't know you have?
 - How can you move to virtualization if you don't know how many servers you are running and what their purpose in your network is?
 - How can you make sure you can protect it and administer it, or, even worse, transform it into a dynamic datacenter?
 - **Solution: Generate an inventory of your network is to work with free tools**
 - **Microsoft Baseline Security Analyzer (MBSA)** to scan their networks for vulnerabilities and required updates
 - Each system's IP address, OS, applications installed & vulnerabilities
 - **Microsoft Visio Connector for MBSA**
 - Turn the results of MBSA into generate a graphical image of your network and display detailed information on each device when you click on it



Computer name: WORKGROUP\STARIELLE
 IP address: 192.168.1.102
 Security report name: C:\DOCUMENTS\Star\LOCAL5\Temp\mp50.xml
 Scan date: 7/5/2006 12:30 PM*** This report is 1 day old.***
 Scanned with MBSA version: 2.0.5029.2
 Catalog synchronization date:
 Security update catalog: Microsoft Update
 Security assessment: Severe Risk (One or more critical checks failed.)

Security Update Scan Results

Score	Issue	Result
X	Office Security Updates	4 security updates are missing. 1 service packs or update rollups are missing. What was scanned Result details How to correct this
✓	Windows Security Updates	No security updates are missing. What was scanned Result details

Windows Scan Results

Administrative Vulnerabilities

Step 1 : Discovery

Scan for Potential Virtualization Candidates

- **Scan for Potential Virtualization Candidates**
 - System assessment
 - **Microsoft - MAP**
 - **VMWare** - VMware Guided Consolidation (VGC)(<100 servers) and Capacity Planner (>100 servers)
 - Microsoft Assessment and Planning (MAP) is an agentless analyzer that collects information on all of the identified systems
 - Database systems that can be migrated to SQL Server 2008
 - Systems and services that can be migrated to Microsoft Online Services
 - Environments that can profit from Network Access Protection (NAP)
 - Power-saving proposals to reduce power consumption in datacenters
 - Systems that can run Vista with Service Pack 1, MS-Office 2007, WS2008
 - Applications that can be virtualized through Microsoft Application Virtualization
 - Performance reports from servers or workstations
 - Server consolidation through virtualization with Microsoft Hyper-V and Windows Server 2008

- VMware Guided Consolidation
 - Analyses the computers in your enterprise and recommends the best candidate for virtualization. Additionally, it guides you through conversion process based on the computers you selected for consolidation. Converting underutilized physical computers can provide cost savings in your environment
 - First, it relies on an agentless infrastructure to discover existing computers and collect performance information from them.
 - Second, it analyzes collected data for hardware consolidation suitability and recommends a potential consolidation plan.
 - It allows you to convert machines from physical to virtual (P2V) installations through the **VMware Converter**.

- VMWare Capacity planner (not a free one)
 - Capacity Planner will give you reports on server processor utilization as well as a comprehensive report that will identify potential virtualization candidates. Analysis results include CPU, memory, network, and disk utilization on a server-byserver basis.



Dashboards

Inventory

Performance

Analyze

Optimize

Administration

Go to System

System Name



Company

VMware

Filter

Domains:

All Domains

Departments:

All Departments

Environments:

All Environments

Functions:

All Functions

Locations:

All Locations

Host Name:

Interval

Hourly ☐ Weekly ☒

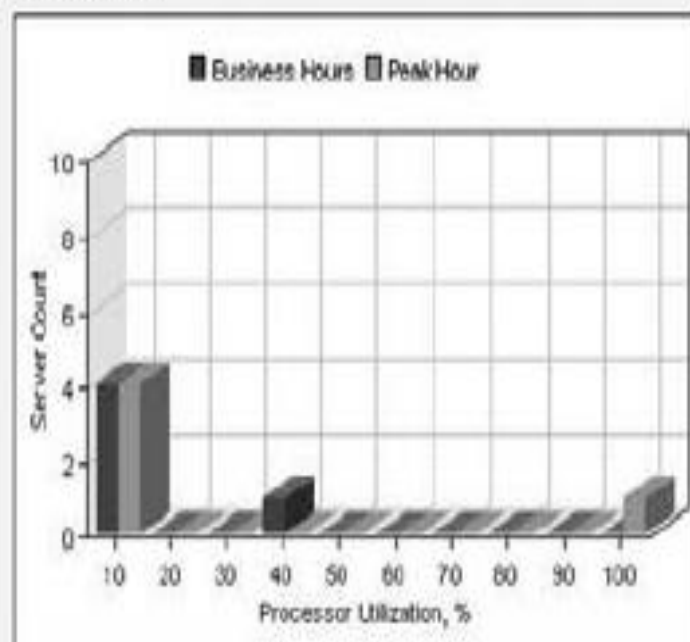
Enterprise Dashboard (Week: 0, Year: 0)

Processor Utilization

Most Recent 12 weeks

Cannot display graph: there are not enough statistics data collected.

Two (2) samples of statistics data from different weeks are required.

Server Distribution by Processor Utilization
Active Servers

Server Count

CPU(s)

No Inventory

1

2

Servers

5911

20

10

Processor Summary

CPU Mhz

Ave CPU Utilization

Peak CPU Utilization

CPUs

7,979

9.99

25.05

4

Optimization Opportunities

Consolidation:

Anomalies:

Alerts:

Trend Deviations:

14

0

0

0

Step 1 : Discovery

Categorize the server resources

- Categorize Server Resources.
 - Once you have the results of your analysis, you can categorize each machine role into groups. For example, you can identify the server roles or functions in the following service types:
 - Network infrastructure and physical servers
 - Identity management servers
 - File and print servers
 - Application servers
 - Terminal servers
 - Dedicated web servers
 - Collaboration servers
 - Servers can be further categorized according to their location and server type
 - Servers can be further categorized according to their Applications or workloads

Step 1 : Discovery
Rationalize Everything

- Rationalize Everything.
 - After you complete the evaluation of the systems you have in place, you will be ready to move on towards virtualization.
 - However, it is important that you distinguish between **server** and **physical consolidation**.
 - **Server consolidation** means reducing the number of servers in your datacenter, creating larger servers that host multiple workloads.
 - **Server consolidation** is in response to the problem of **server sprawl**, a situation in which multiple, under-utilized servers take up more space and consume more resources than can be justified by their workload
 - Yet in most cases, organizations that perform a virtualization project will not perform server consolidation; they will perform physical consolidation.

- Rationalize Everything.
 - Organization A has 100 physical servers. After performing an extensive analysis and selecting the virtualization engine to use, Organization A determines that they can transform each and every existing workload into a virtual machine. To do so, they must **use 10 physical machines to run the 100 virtualized workloads**. The result? **One hundred ten servers**.
 - This is not a server consolidation project. **It is a physical consolidation project**.
 - Organization A will profit from the move to virtualization because they will have 90 fewer machines to manage, but their administrators will not have less work since they now have to manage 10 more machines than before
 - This is why rationalization is the most important aspect of any virtualization project.
 - Your **goal should be to have fewer servers** when you finish the virtualization project than when you began, and this should include the physical servers you will need to run the virtual machines you create. There is no better time to perform a **service or workload consolidation or rationalization** than when you are migrating to a new virtualized infrastructure. .

Step 2: Virtualization

Step 2 - Virtualization

Seven layers of virtualization

**Server
Virtualization
(SerV)**

**Storage
Virtualization
(StoreV)**

**Network
Virtualization
(NetV)**

**Management
Virtualization
(ManageV)**

**Desktop
Virtualization
(DeskV)**

**Presentation
Virtualization
(PresentV)**

**Application
Virtualization
(AppV)**

Step 2 - Virtualization

Server Virtualization (SerV)

Server Virtualization (SerV) is focused on **partitioning a physical instance of an operating system into a virtual instance** or virtual machine.

There are two aspects of server virtualization:

- **Software Virtualization (SoftV)** runs the virtualized operating system on top of a software virtualization platform running on an existing operating system.

- **Hardware Virtualization (HardV)** runs the virtualized operating system on top of a software platform running directly on top of the hardware without an existing operating system. **The engine used to run hardware virtualization is usually referred to as a hypervisor.**

Step 2 - Virtualization

Server Virtualization (SerV)

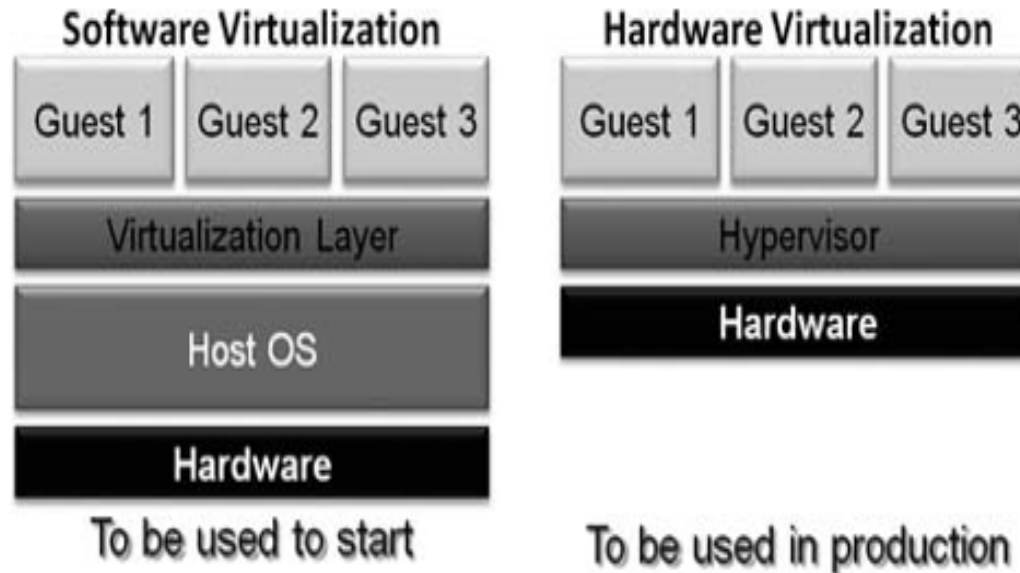
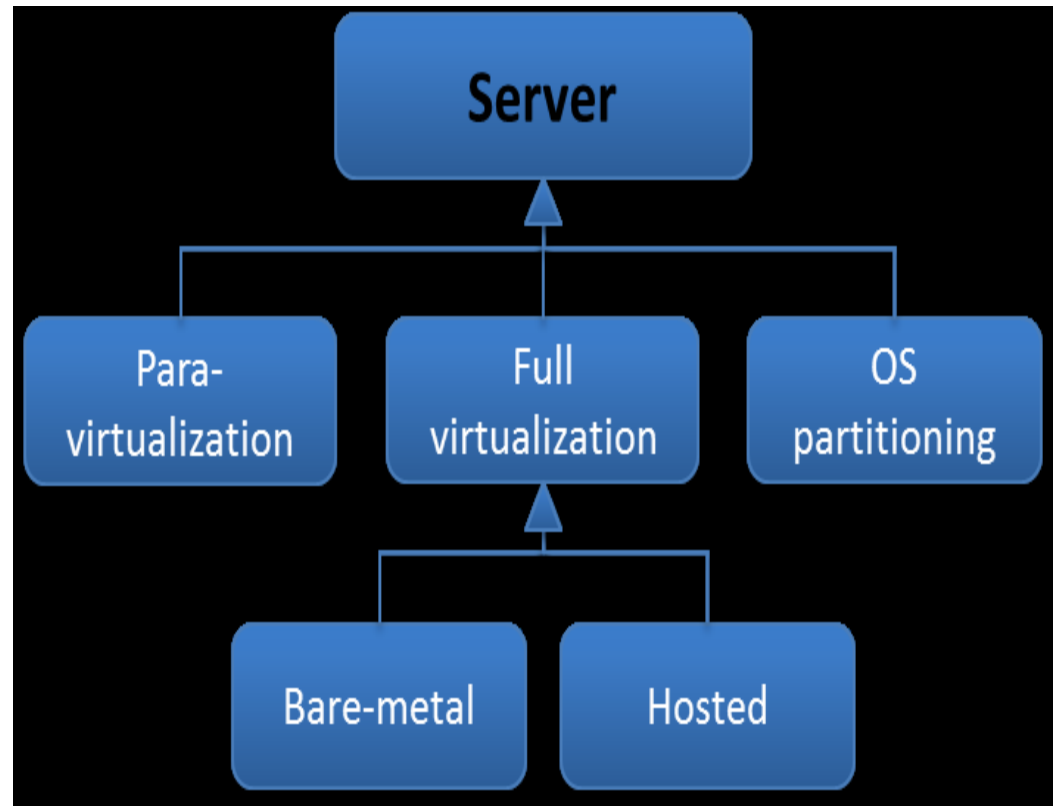


Figure 2-10. The two different server virtualization models

Server Virtualization

- Server virtualization can be divided in 3 types or sub classes: **Para-virtualization**, **full virtualization** and **OS partitioning**.
- Full virtualization technologies can be divided into two more sub classes: **“Type 1”** and **“Type 2”** hypervisors



Major Server Virtualization Vendors

- ☐ Citrix
- ☐ Microsoft
- ☐ VMware
- ☐ Oracle
- ☐ Novell
- ☐ Red Hat
- ☐ IBM
- ☐ Sun

Benefits of server virtualization

- The first one is certainly at the deployment level. A virtual machine can often be built and customized in less than 20 minutes.
 - You can deliver a virtual machine that is ready to work right away in considerably less time than with a physical machine.
- Another benefit is virtual machine mobility. You can move a VM from one host to another at any time. In some cases, you can move it while it is running.
 - This is a great advantage and will help reduce downtime in your network.
- Once it is built and configured, you just start the machine and it is immediately ready to deliver services to users.
 - Virtual machines are just easy to use.
- Virtual machines support standard configurations. You can control the way VMs are built: Just create a standard VM and copy the source files for this VM each time you need a new machine.
 - This way, you will always have standard configurations for any VM.

Benefits of server virtualization

- Virtual machines also support the concept of volatile services. If a tester or developer needs a virtual machine to perform a given series of tests, you can fire up a new VM, provide it to them in minutes, and then, when they are done with it, you simply delete it. Try doing that with a physical machine!
- VMs can be certified by the virtualization vendor, ensuring you are using the best of their technology's capabilities with your VMs.
- VMs are also secure because they can be completely isolated at any time; just cut off their communications through the host's virtualization technology.
- VMs can be scaled out or scaled up. To scale out, simply create more VMs with the same services. To scale up, shut down the VM and assign more resources, such as RAM, processor cores, disks, and NICs to it.
- VMs are also ideal for disaster recovery, since all you need to do is copy their files to another location, either within your datacenter or to another site entirely.

Step 2 - Virtualization

Storage Virtualization (StoreV)

- Storage Virtualization (StoreV) is used to **merge physical storage from multiple devices so that they appear as one single storage pool.**
- The storage in this pool can take several forms: direct attached storage (**DAS**), network attached storage (**NAS**), or storage area networks (**SANs**).
- Though storage virtualization is not a requirement for server virtualization, one of the key strengths you will be able to obtain from storage virtualization is the ability to rely on thin provisioning or the assignation of a logical unit (LUN) of storage of a given size, but provisioning it only on an as-needed basis.

Step 2 - Virtualization

Storage Virtualization (StoreV)

For example, if you create a LUN of 100 gigabytes (GB) and you are only using 12GB, only 12GB of actual storage is provisioned. **This significantly reduces the cost of storage since you only pay as you go.**

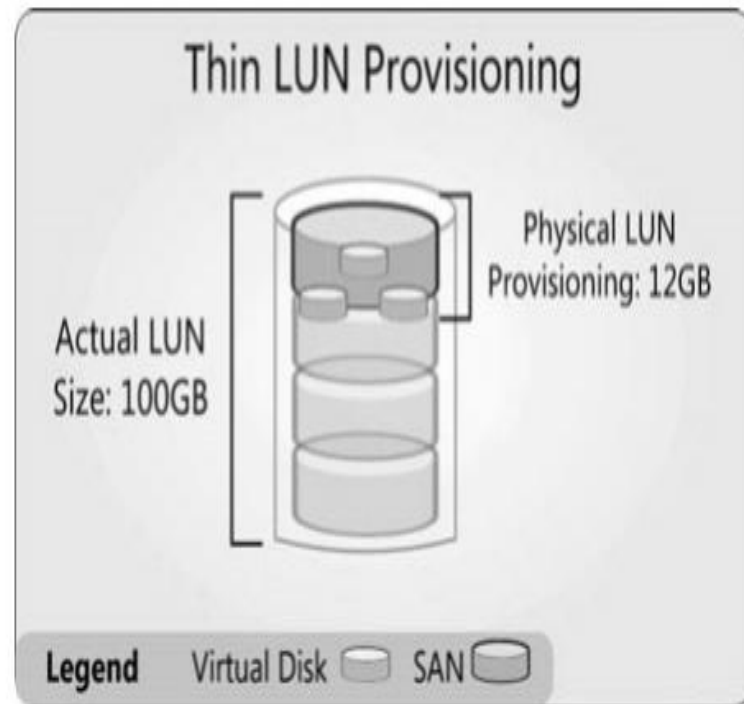
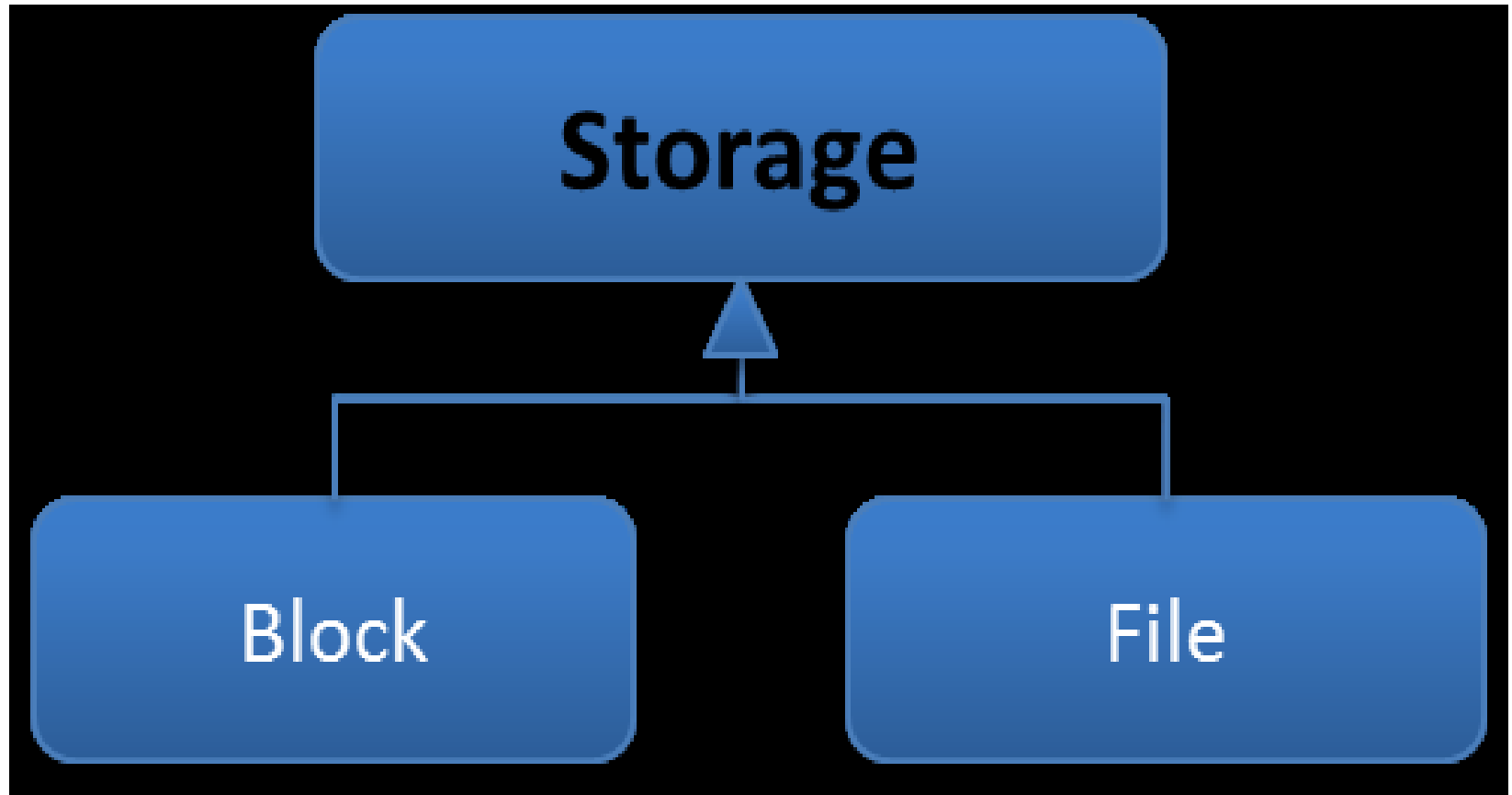


Figure 2-8. Using thin provisioning through Storage Virtualization can significantly reduce storage costs.

Storage Virtualization



Block-Level

File-Level

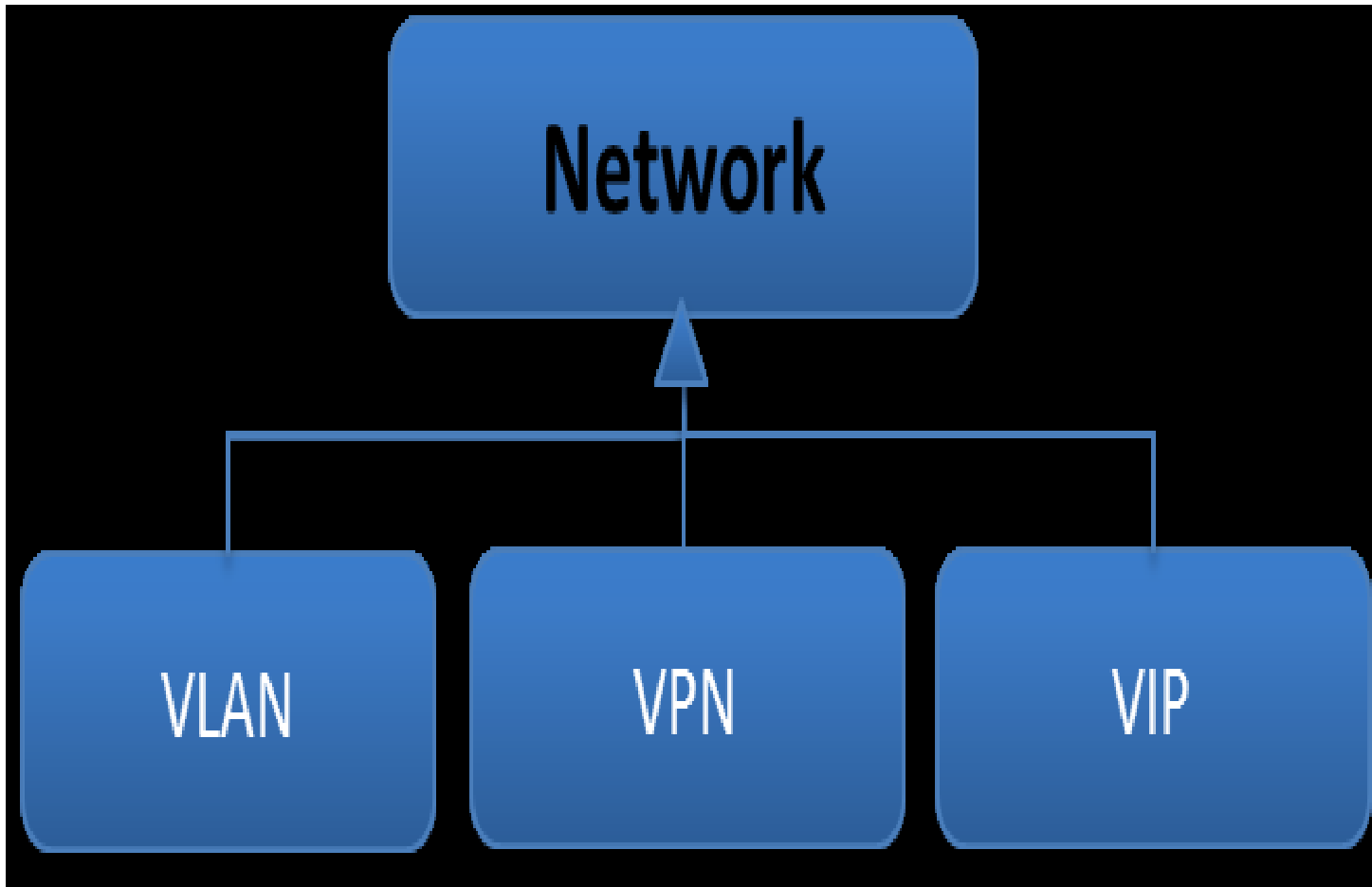
Protocols	Fibre Channel/ iSCSI	Samba/ CIFS
Applications	<p><i>A LUN can be targeted as a physical drive.</i></p> <p>It can be a ... <i>Logical Disk</i> <i>operation system</i> <i>file storage</i> <i>database</i></p> <p>You can <i>format it</i> <i>install OS/applications on it</i> <i>boot-up of systems</i> <i>build file-system (NFS, NTFS, SMB, VMFS)</i></p>	<p><i>A shared folder can be mounted as a network drive.</i></p> <p>It can be a ... <i>file storage</i></p> <p>You can <i>set access privilege</i> <i>integrate with corporate directories</i></p>
Pros	<p><i>Data transportation is much more efficient and reliable.</i></p> <p><i>Support individual formatting of file systems.</i></p>	<p><i>Simple to use and implement.</i></p> <p><i>Inexpensive to maintain.</i></p>
Data Deduplication	<p>Block 1 Block 2 Block 3 Block 4 Block 5</p>  <p>* It saves network bandwidth usage, but may require additional time for comparing data.</p>	<p>File 1 File 2 File 3</p> 
Incremental Backup	 <p>Day 2 Backup</p> <p>Day 1 backup</p>	 <p>Day 2 Backup</p> <p>Day 1 backup</p>

Step 2 - Virtualization

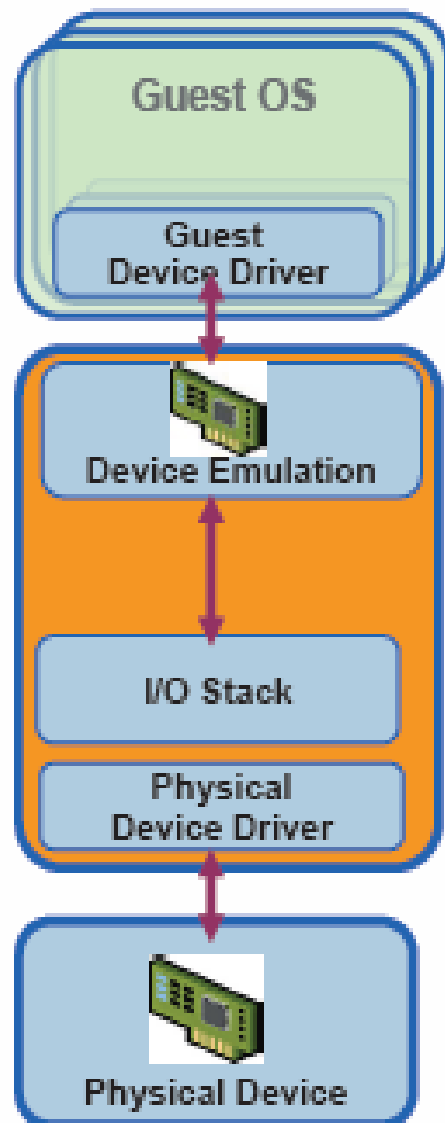
Network Virtualization (NetV)

- Network Virtualization (NetV) lets you **control available bandwidth by splitting it into independent channels** that can be assigned to specific resources.
- For example, the simplest form of network virtualization is the **virtual local area network (VLAN)**, which creates a logical segregation of a physical network.
- In addition, server virtualization products support the creation of virtual network layers within the product itself.
 - using this **virtual network layer** would let you place a perimeter network on the same host as other production virtual workloads without impacting either of the networks or letting the virtual machines access each other.

- Types of NetV
 - Virtual LAN (VLAN),
 - Virtual IP (VIP) and
 - Virtual Private Network (VPN).

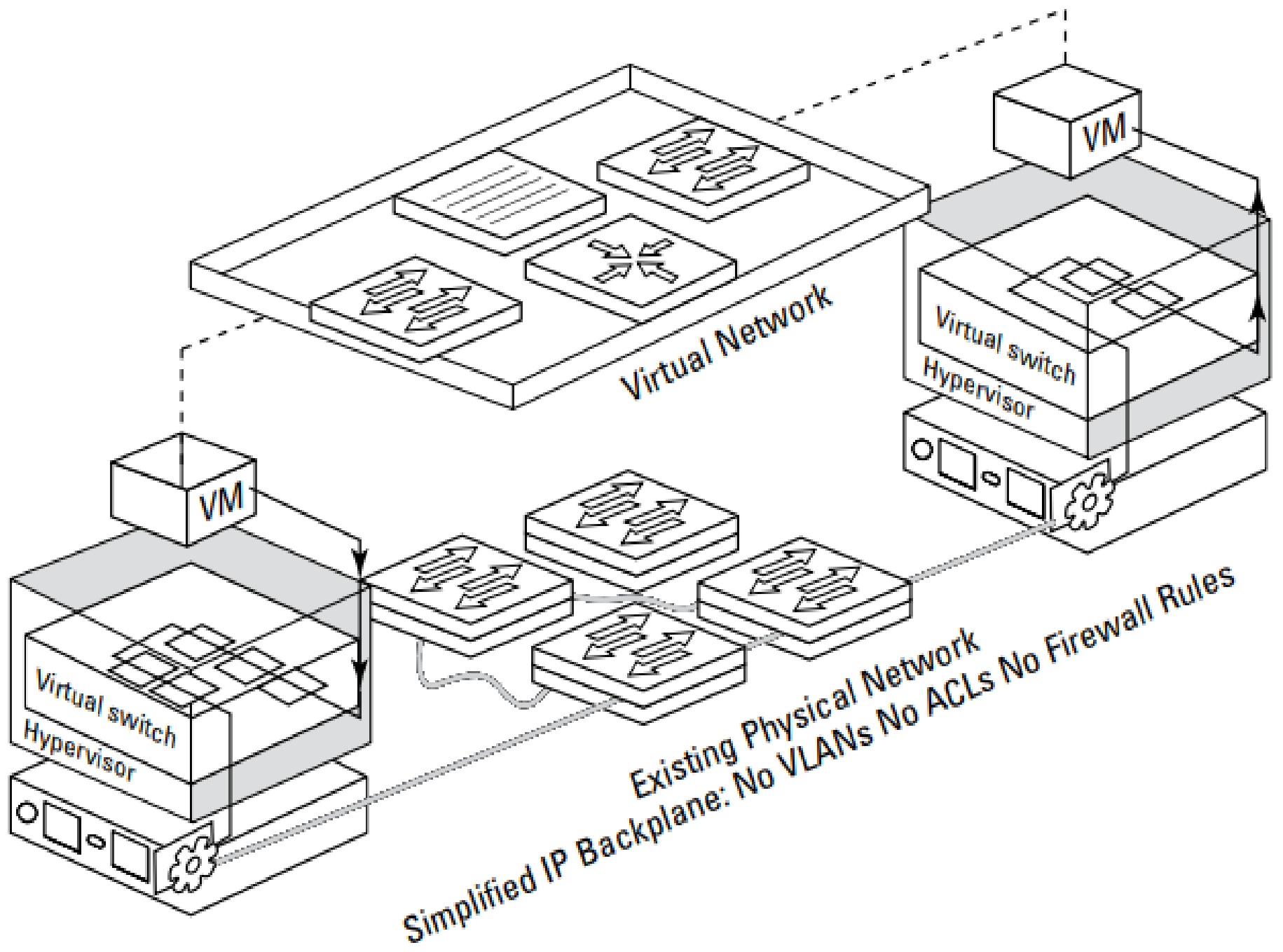


I/O Virtualization Architecture



○ I/O Virtualization architecture consists of

- Guest driver
- Virtual device
- Communication mechanism between virtual device and virtualization stack
- Virtualization I/O stack
- Physical device driver
- Real device



Step 2 - Virtualization

Management Virtualization (ManageV)

- Management Virtualization (ManageV) is focused on the technologies that manage the **entire datacenter, both physical and virtual, to present one single unified infrastructure** for the provision of services.
- ManageV is not necessarily performed through a single interface.
- In fact, you should make sure that two key layers are segregated at all times:
 - **Resource Pools (RP)**, which includes the collection of hardware resources— host servers, racks, enclosures, storage, and network hardware—that makes up the datacenter infrastructure
 - **Virtual Services Offerings (VSO)**, or workloads that are made up of the virtual machines—servers and/or desktops—that are client-facing and offer services to end users .

Step 2 - Virtualization

Desktop Virtualization (DeskV)

- Desktop Virtualization (DeskV) allows you to rely on virtual machines to **provision desktop systems**.
- Desktop virtualization has several advantages, the least of which is the **ability to centralize desktop deployments and reduce distributed management costs** because users access centralized desktops through a variety of thin or unmanaged devices.
 - **Client** desktop virtualization technologies are used to host virtual desktops (or virtual machines) locally on the clients' computer.
 - Server desktop virtualization -two types: personal & shared.
 - Shared desktops -shared among users
 - Personal desktops -users having their own completely isolated desktop.

Desktop

```
graph BT; Client[Client] --> Desktop[Desktop]; Server[Server] --> Desktop; Personal[Personal] --> Server; Shared[Shared] --> Server; Virtual[Virtual] --> Personal; OV[Offline Virtual] --> Personal; Physical[Physical] --> Personal;
```

Client

Server

Personal

Shared

Virtual

**Offline
Virtual**

Physical

Step 2 - Virtualization

Presentation Virtualization (PresentV)

- Presentation Virtualization (PresentV), until recently called Terminal Services, **provides only the presentation layer** from a central location to users.
- While the need for PresentV is diminishing because of the introduction of technologies such as Application Virtualization, the protocols used for PresentV are at the forefront of both DeskV and SerV technologies since they are the protocols used to access, use, and manage virtual workloads.

Step 2 - Virtualization

Application Virtualization (AppV)

- Application Virtualization (AppV) uses the same principles as software based SerV, but instead of providing an engine to run an entire operating system, AppV **decouples productivity applications from the operating system.**
- AppV transforms the **distributed application management model** because you only need to **virtualize an application once.** From then on, the **application virtualization engine will make the virtualized application run on any version of Windows.**

Step 2 - Virtualization

While there are seven layers of virtualization, there are other key terms that make up the language of virtualization in the datacenter.

- **Host server** - The physical server running virtual machine workloads.
- **Guest operation system** - A virtualized operating system running as a workload on a host server.
- **Resource Pool** - The collection of hardware resources, including host servers that make up the datacenter infrastructure.
- **Virtual Service Offerings** - The virtual machines that are client-facing and offer services to end users. They are also often referred to as virtual workloads.
- **Virtual Appliances (VAPs)** - Pre-packaged VSOs that run a specific application or workload.
- **Policy-based workloads** - VSOs that are powered up on an as-needed basis through automated policies.
- **Operating system virtualization** - Often misconstrued as guest OS virtualization, this is nothing more than OS partitioning because it can only run one single OS type in parallel instances.

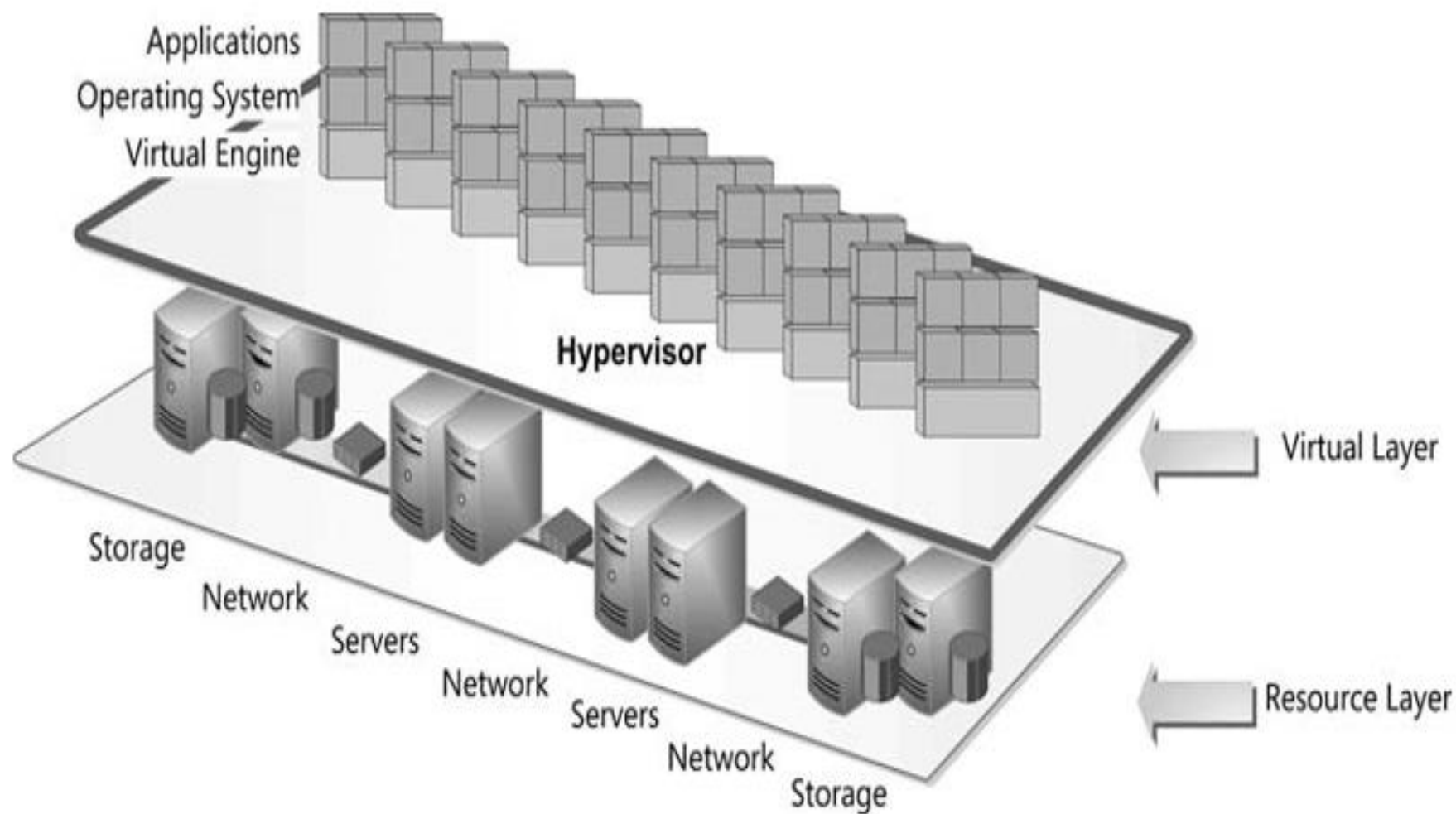


Figure 2-11. Working with resource and virtual layers in the datacenter

Step 3 – Hardware Maximization

- When you move your servers and desktops to virtual workloads, you must consider **changing the nature of the hardware** you rely on to ensure that the systems you run your virtual workloads on are **always highly available**.
- In addition, since the transformation of the datacenter relies on moving low-use workloads onto fewer hardware systems, it is important to make sure that each of the **systems you run offers the widest amount of resources to these workloads**.

Step 3 – Hardware Maximization



Move to 64-Bit

The diagram consists of four horizontal rows. Each row features a blue rounded rectangle on the left containing text, and a white rectangle on the right. A thin blue line connects the right side of the blue rectangle to the left side of the white rectangle. The blue rectangles are stacked vertically with consistent spacing between them.

Rely on Shared Storage

Be Aware of Licensing Costs

Rely on New Server Form Factors

Step 3 – Hardware Maximization

Move to 64-Bit

- Since your hardware servers are now going to run multiple virtual workloads, you must consider hardware that can deliver the best price/performance. One of the first places to start is with potential bottlenecks.
- One of the most important bottlenecks is random access memory (RAM).
- Since each virtual machine must address its own memory, your server configurations must be designed with the utmost in memory configurations.
- In addition, the **system must use the best resources to make this memory available to each virtual machine.**
- For this reason, it is important to **consider 64-bit processor architectures.**

Step 3 – Hardware Maximization

Rely on Shared Storage

- In addition to removing resource bottlenecks in your server configurations, you must remove potential single points of failure. When you place 10, 20, or even 30 virtual machines on a single host server, you must make sure that there is no way for this host server to fail; otherwise, all of the virtual workloads will also fail, causing some irate end users.
- For this reason, you must **configure hardware with high availability** in mind.
- Several computer nodes are linked together to protect the workloads they run. Should one machine fail, the workloads are automatically transferred to another available node.
 - What makes these configurations work is **shared storage**.

Step 3 – Hardware Maximization

Rely on Shared Storage

- Each node is linked to the **same storage container**.
- This means that when a workload needs to fail over, the files that make it up do not need to be moved or copied to the next available computer node.
- In addition, virtualization technologies support the ability to dynamically move workloads from one system to another when the virtual workload needs more resources than are currently available on the node it resides on at the time.
- The **workload is automatically moved to another node that has available resources without administrative interaction**.
- Once again, this can only be made possible when the **virtual workload is located on shared storage** of some sort.

Step 3 – Hardware Maximization

Rely on Shared Storage

- You can store **all virtual machines on a shared storage subsystem**.
- In addition, because the systems attached to a shared storage infrastructure are located in the **same site**, they are considered to **provide site-level redundancy**.
- And, since the virtual machine is the point of failure, every application you run inside of them is automatically highly available.
- **Be careful that your shared storage subsystem does not become a single point of failure of its own.**
- **Always think redundancy** when planning for hardware components that will support virtual workloads.

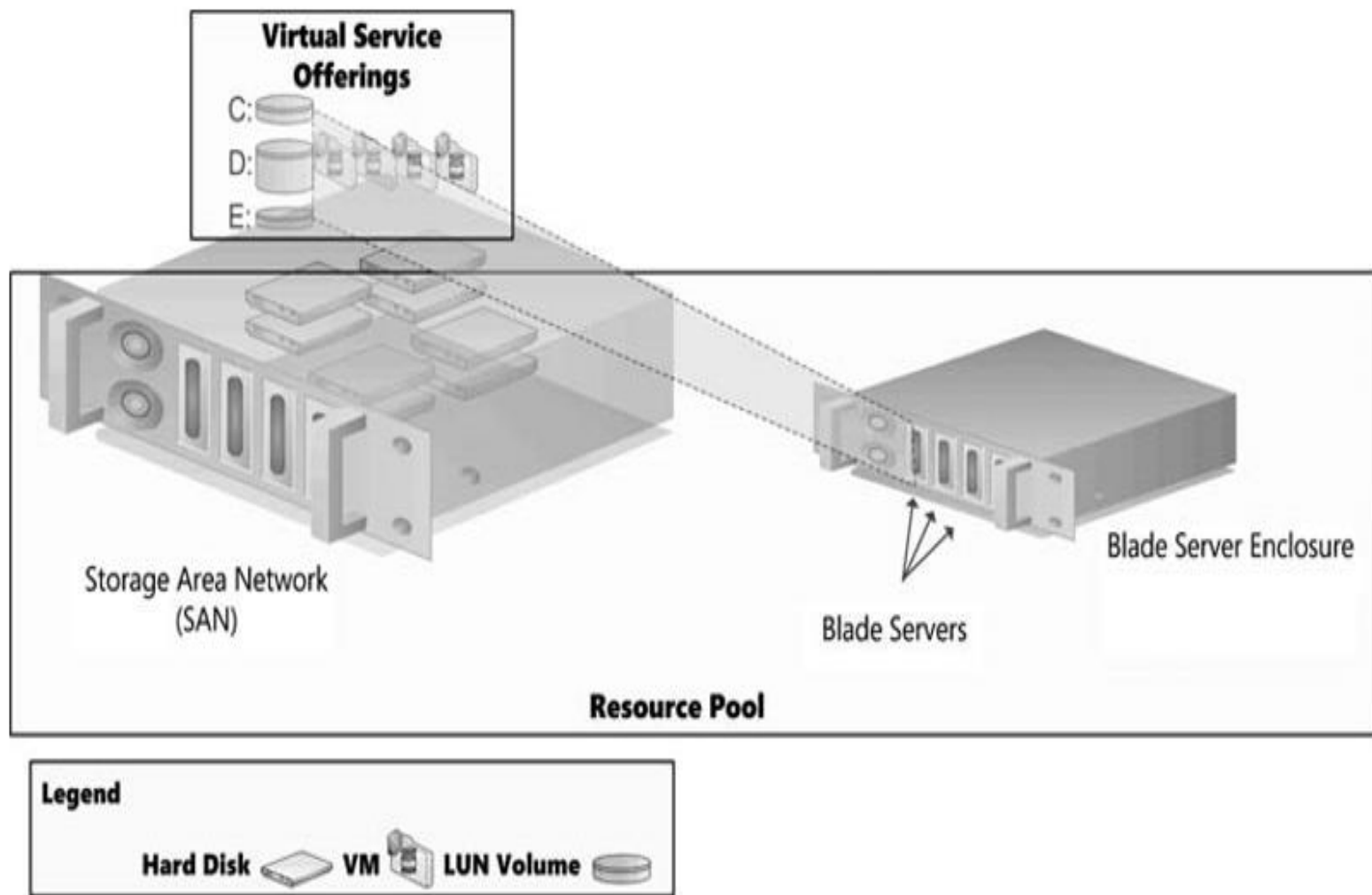


Figure 3-3. Using shared storage allows you to create highly available configurations.

Step 3 – Hardware Maximization

Be Aware of Licensing Costs

- The monetary savings of virtualization are incredible and make for an easy business case.
- First, you save on all of the hardware you no longer need to buy.
- Second, software manufacturers such as Microsoft have changed their software licensing models to support virtualization ever since the release of Windows Server 2003 R2 (WS03R2) and continuing on with Windows Server 2008 (WS08).
 - Buying a license of Windows Server 2003 R2 or Windows Server 2008 Enterprise Edition now grants you up to **four free virtual instances of the OS**, as well as **providing a license for the host server** if you choose to run Windows Server on it.
 - Each **additional license gives you up to four more guests**. Buying a license of Windows Server 2003 R2 or **Windows Server 2008 Datacenter Edition gives you the right to run an unlimited number of virtualized instances** of any Windows Server OS

Step 3 – Hardware Maximization

Rely on New Server Form Factors

- Today, organizations build complex multipurpose servers for remote offices or small organizations. With virtualization, you no longer need to do this.
- For example, you might build a remote office server to include several **roles—domain controller, file server, print server, management server, database server, and so on.**
- Building such a server requires many compromises.
 - Domain controllers should always be protected at all times. Having users access file or print services through a domain controller forces you to either reduce the security of the server or make file access more complex than it needs to be. **With the advent of virtual machines, you can now separate all of these roles without having to pay extra licenses.** But this means looking to new server types. **All-in-one-servers or server pods, which include two physical servers, network switches, and shared storage are ideal for this scenario,** whether it is for a remote or small office.

Step 4 – Architectures

- The next step in the process is to update and revise your datacenter architectures.
- Working with multiple virtualization layers can become quite confusing if they aren't structured properly.
- You've already seen that there are quite a few elements to a virtualized infrastructure.
- For example, to protect your virtual workloads, you'll want to **use x64 systems linked to shared storage and arranged into some form of high-availability cluster.**
- But you can also cluster in the virtual layer.
- For example, when you build an Exchange Server 2007 Mailbox server to store user mailboxes, you'll most likely want to take advantage of the failover clustering and replication capabilities of this product to ensure that the Mailbox service provided by the Virtual Service Offering is highly available.

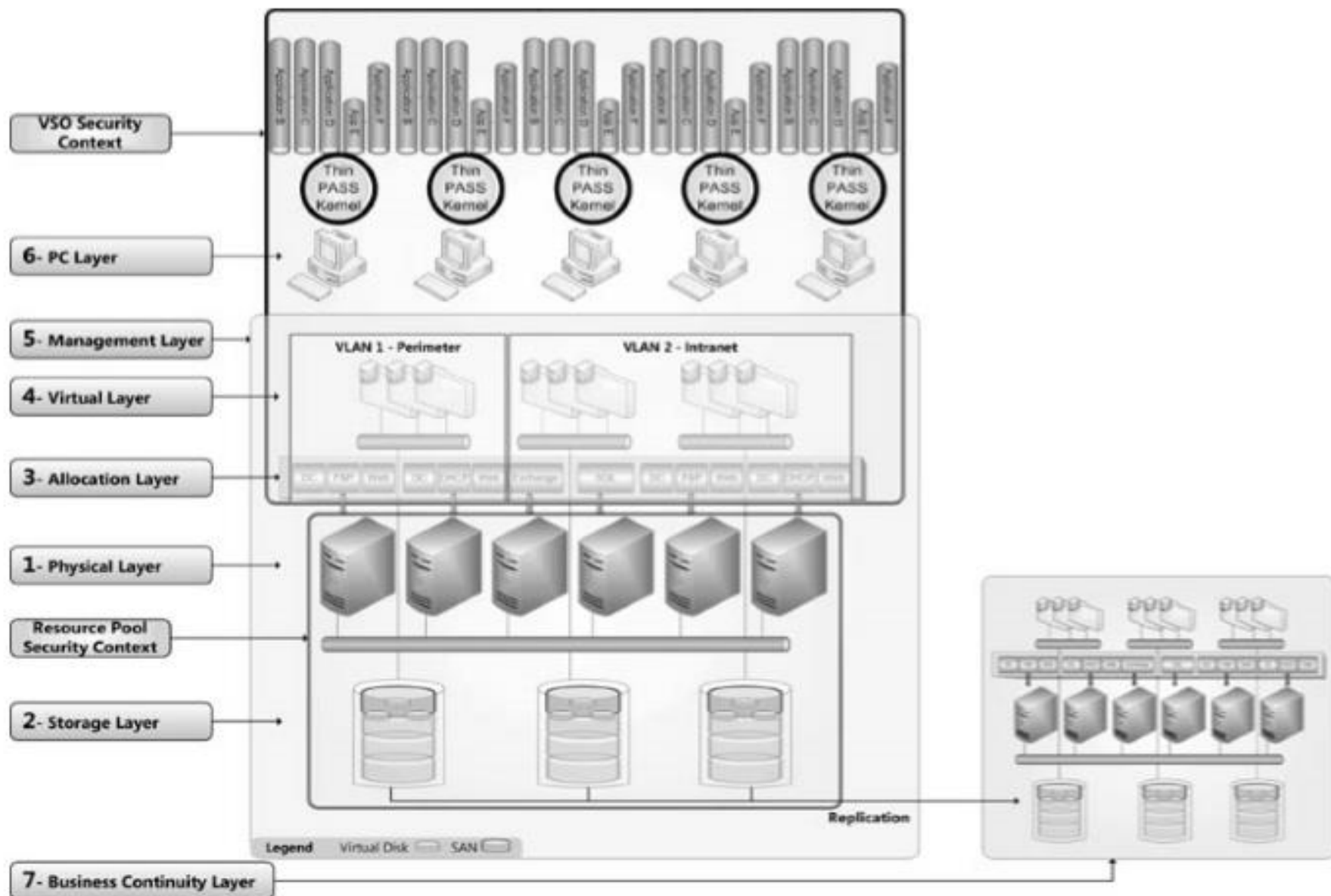


Figure 3-9. Integrate virtualization layers into one single-service architecture.

Step 5 – Manage Virtualization

Focus on VM Management

- You will need to ask the right questions when you are looking for new management to:
 - How are VM source images tracked and updated?
 - How are VM snapshots tracked?
 - How are move, adds, and changes (MACs) tracked, especially when machines are dynamically moved from one host to another?
 - How are virtual workload policies prepared and deployed?
 - Are there special dependencies between applications and VMs?
 - How is security managed between resource pools and Virtual Service Offerings? What kind of segregation is in place? Are all communications to resource servers encrypted to protect passwords?
 - What kind of troubleshooting support does the tool provide when VMs or hosts go down? Can the tool identify root causes—physical or virtual or both—of the problems you face?
 - Is the tool fully functional and does it provide a one-stop shop? Can it manage every aspect of virtualization?

Step 5 – Manage Virtualization

Rely on Virtualization Decision

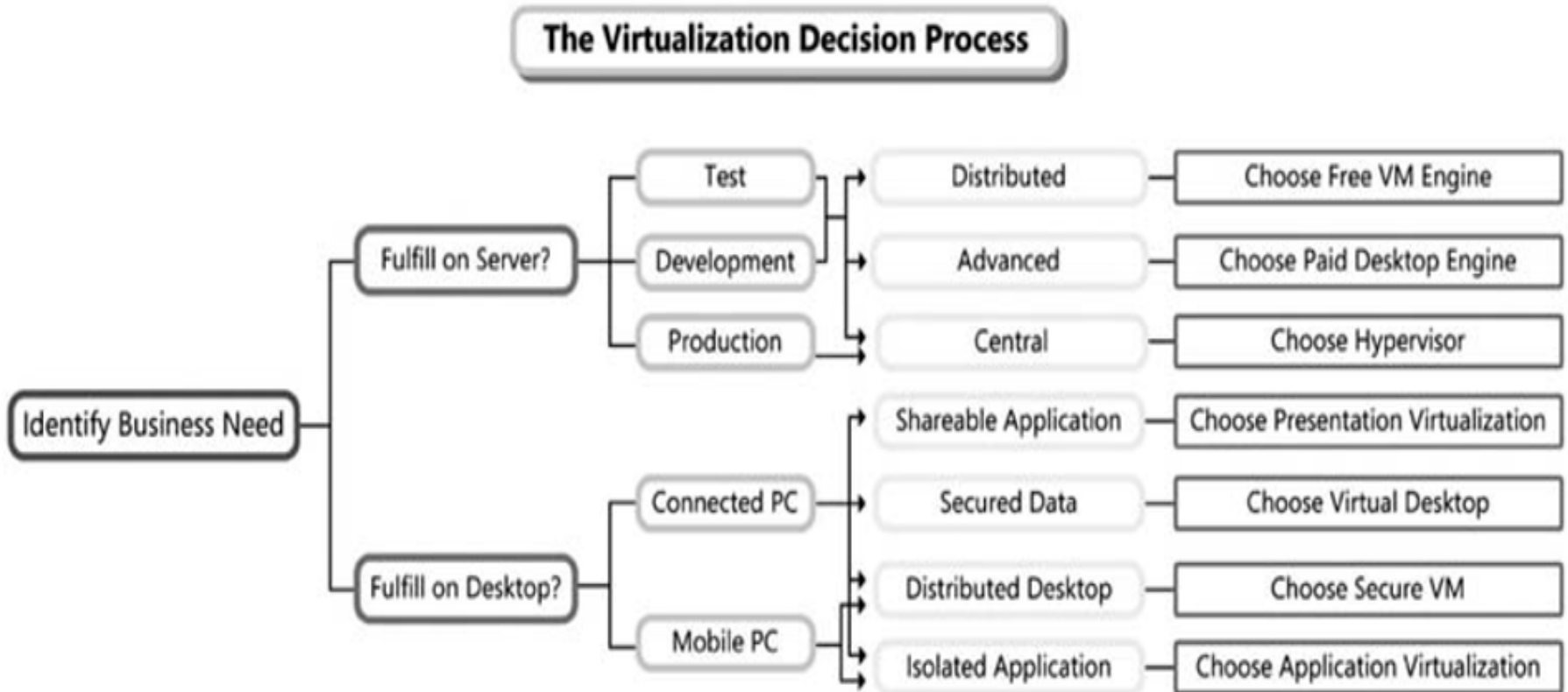
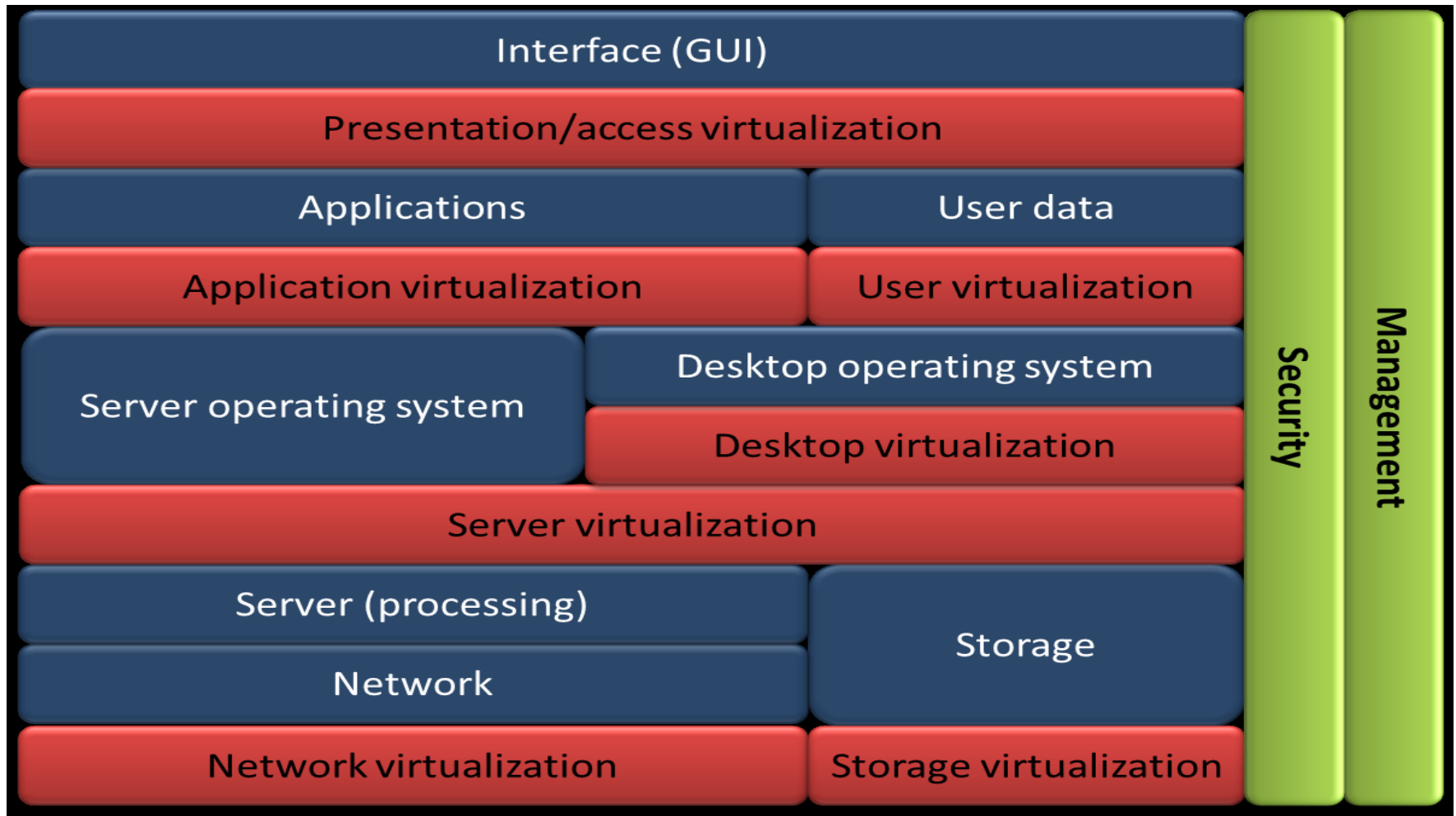


Figure 3-12. The virtualization decision process

Seven Layers of Virtualization



The blue layers

- The blue layers indicate the hardware and software layers that are present in a typical server environment.
- **Network layer** -concerned with the network requests of the upper layers and facilitates the communication from server to server or storage device, which can be located on the same private network or on a public network, such as the Internet.
- It is responsible for controlling the network operations imposed by the server and operating system in the upper layers by delivering or transferring data from source to destination.

The blue layers

- **Storage layer** -concerned with the storage and retrieval of data.
- The storage layer is put next to the server (processing) layer as well as the network layer, because storage can be a storage device attached to the server via a network (SAN and NAS) or local storage directly attached to the server.
- **Server (processing) layer** -concerned with carrying out the instructions of operating system layer above.
- The server consists of hardware components that process the instructions imposed by the upper layers, such as a Central Processing Unit (CPU), memory, or a Network Interface Card (NIC).

The blue layers

- ❑ **Server and desktop operating system layer** act as intermediaries with applications and computer hardware.
- ❑ It is a software platform that manages the applications of the upper layer and handles the request of the computer programs or applications.
- ❑ Regulates the ways applications communicate with computer hardware and the way users interact with the other layers in the model.
- ❑ However, **in a traditional server environment without virtualization, a server operates one type of operating system (OS) or instance at a time.**

The blue layers

- **With virtualization, which will be explained in the red layers below, multiple instances or operating systems with their own applications can operate at the same time.**
- Each instance is called a virtual machine.
- This way, it is also possible to host multiple virtual machines with pre-configured desktops, next to other virtual machines that host a web server or database.
- These virtual machines are called **virtual desktops**.
- Therefore, in a virtual environment multiple (desktop or server) operating systems can exist next to each another.

The blue layers

- **Application layer** -concerned with requests and tasks imposed by the interface layer above.
- The applications in the layer are designed to perform an activity or set of tasks that can be performed when a request or action is made in the interface layer above.
- **User data layer** - concerned with the actions and changes made in the application layer.

The blue layers

- In a desktop environment, users can perform tasks such as installing new applications, making or editing documents, changing settings to their preference, etc.
- These tasks produce new personal data, which are called user data.
- **Interface layer** is the point of interaction between a user and the lower layers.
- The Graphical User Interface (GUI) allows people to interact with computer programs in the layers below by offering graphical icons and elements which can be controlled.

The red layers

- The red layers in the taxonomy illustrate the levels and layers in which the main virtualization domains relate to each other.
- On the bottom, **network virtualization and storage virtualization** can be applied in the network layer and the storage layer.
- For network and storage virtualization technologies it is not a prerequisite to have server virtualization or other types of virtualization technologies.
- **Server virtualization** takes place above the server (processing) and storage layer and beneath the operating system layer.
- This layer is also called the virtualization layer, which enables the creation of multiple isolated instances called virtual machines.

The red layers

- ❑ Each virtual machine is equipped with their own OS and application(s), which means that multiple operating systems can operate at the same time.
- ❑ Server virtualization comes also equipped with network and storage virtualization technologies.
- ❑ Furthermore, with server virtualization it is also possible to create virtual desktops, which is another term for a virtual machine intended for desktop purposes.
- ❑ In the same manner as virtual machines, multiple virtual desktops can be placed on a server, which means that multiple desktop operating systems can be placed on top of the server virtualization layer.

The red layers

- **Desktop virtualization** is placed directly on top of the server virtualization layer.
- A virtual desktop is a virtual machine.
- Depending on the type of desktop virtualization technology, virtual desktops can be made for each user or a (shared) virtual desktop for a group of users.
- **Application virtualization** takes place above the operating system layer and allows for the decoupling of the applications layer from the operating system layer.
- This way, applications can be isolated from the OS and can be executed without installation on the OS.

The red layers

- Application virtualization technologies have been used without other virtualization domains, such as server virtualization and desktop virtualization.
- However, full application virtualization requires presence of a virtualization layer or server virtualization.
- Full application virtualization means using the full palette of application virtualization technologies, where applications are streamed or executed from a central location.
- This way, applications can be updated or changed from one point in a central location.

The red layers

- **User virtualization**, also known as user state or profile virtualization creates an abstraction layer between user data by separating user data, such as profile settings, user application data and storage data from OS and application.
- The user data is now stored separately from the application or OS and not interwoven.
- **Presentation or also called access virtualization** creates an abstraction layer between user interface and the other layers below, which allows to user to access and control the other layers remotely.
- The General User Interface (GUI) is streamed towards the device. User virtualization and presentation or access virtualization are both virtualization domains that in particular are used in combination with desktop virtualization.

The green layers

- The green layers show the relation between the management and security domain with the other main virtualization domains.
- Both **management and security tools** are also seen as types of virtualization technologies, because with virtualization new security and management issues have been addressed.
- This has led to the development of new management and security tools for virtual environments.
- Management and security tools have a relation with each of the other layers and have a supporting function.
- For each particular layer, management and security tools can be used to provide management or security measures.

Virtual Machines and Virtual Machines basics

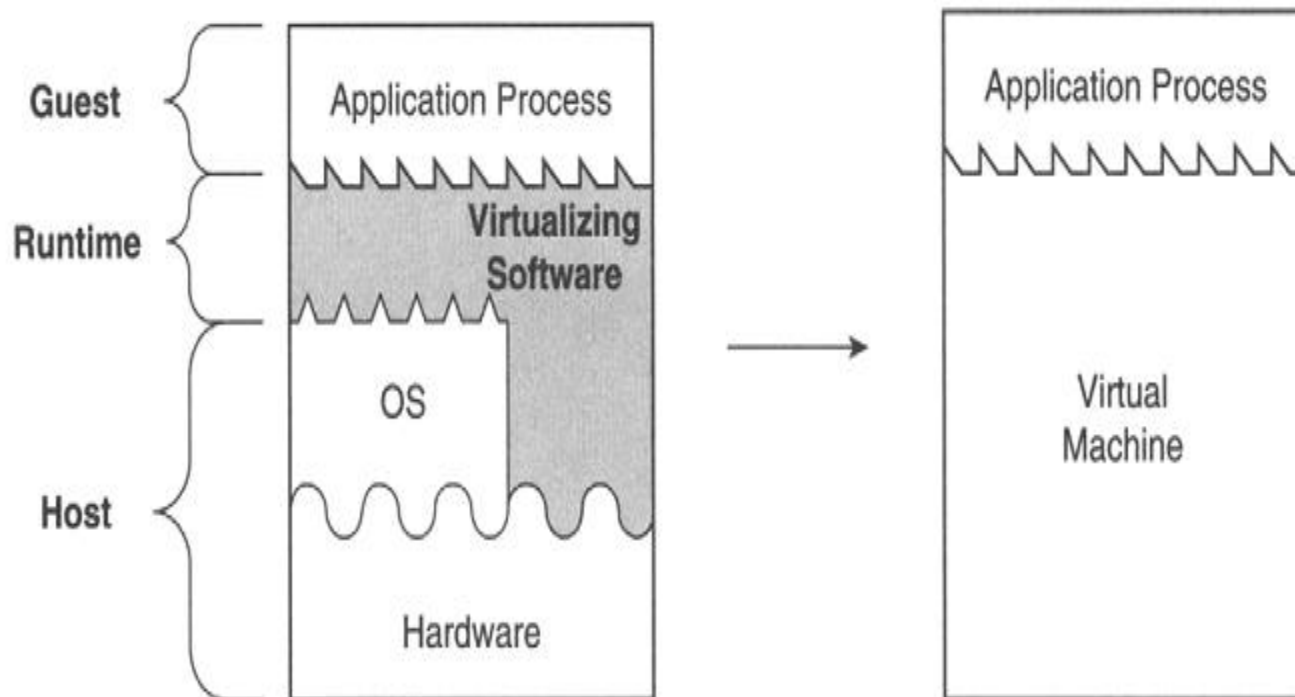
- It is a large umbrella of technologies and concepts.
- Virtualization levels :-
 - Hardware level
 - Operating system level
 - Programming Language level
 - Application level
- Types :- Hardware , Software & Networking

Types of Virtual machines

- Process Virtual Machine
- System Virtual Machine

Process Virtual Machines

Virtualizing software translates a set of OS and user-level instructions composing one platform to another, forming a process virtual machine capable of executing programs developed for a different OS and a different ISA.



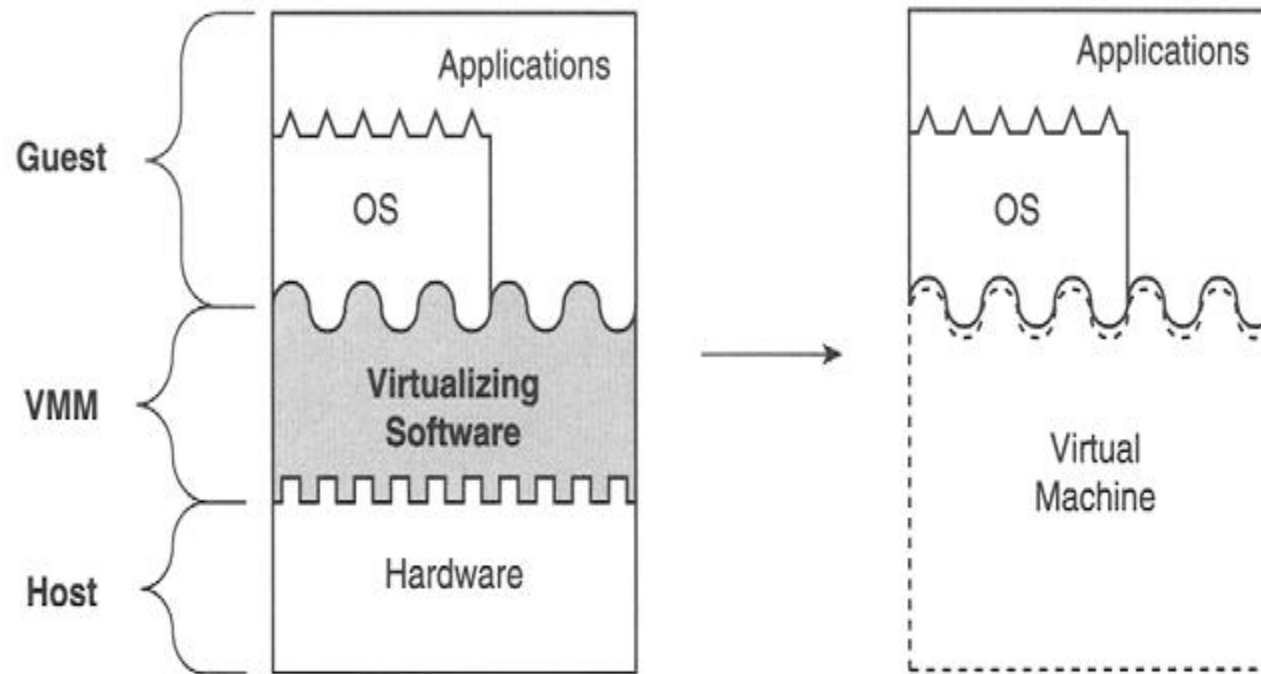
Continued ...

- Supporting individual processes
- Virtualization software is placed over the ABI
- Host – underlying platform
- Guest – software in VM
- Native – platform in the VM
- Virtualizing software – runtime that supports the guest process and runs on top of operating system.

System VM

- Provides a complete system environment.
- Supports an operating system along with its potentially many user processes.
- Provides a guest operating system with access to underlying hardware resources, including networking, I/O, a display and graphical user interface.

Continued ...



VM Basics

- Virtual Machines

- Virtual machines are the fundamental components of virtualization. They are the containers for traditional operating systems and applications that run on top of a hypervisor on a physical server

- **Describing a Virtual Machine**

- The main files that make up a VM are the **configuration file** and the **virtual disk** files.
- The **configuration file** describes the resources that the VM can utilize: it enumerates the virtual hardware that makes up that particular VM.
- The **Virtual disk file** is the secondary storage of your VM where the OS and all the user applications are installed.

- **Examining CPU in a Virtual Machine**

- The single CPU VM can schedule a single CPU's worth of capacity. The host does not reserve a CPU solely for the use of a particular VM; instead, when the VM needs processing resources, the hypervisor takes the request, schedules the operations, and passes the results back to the VM through the appropriate device driver.

- **Examining Memory in a Virtual Machine**

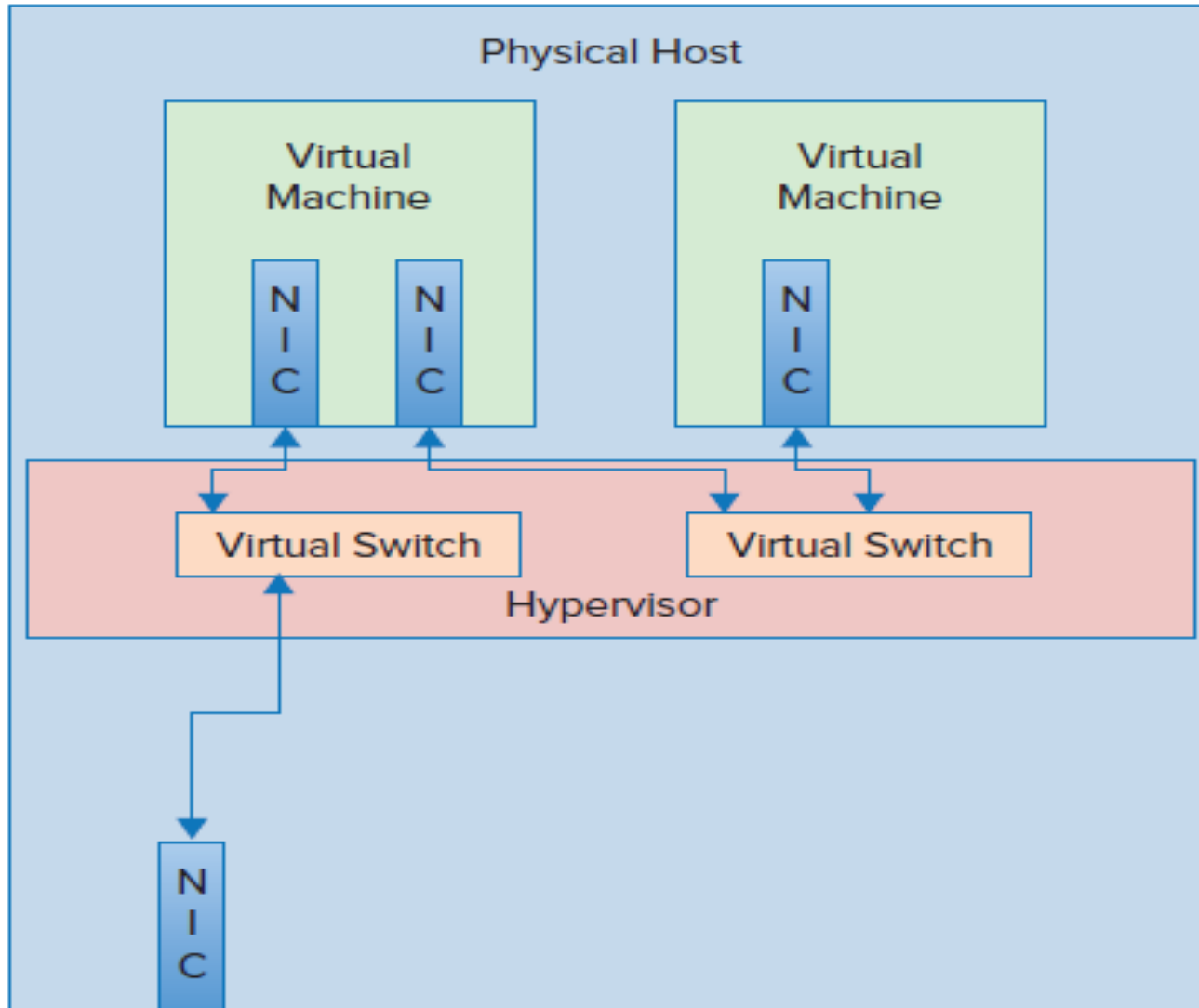
- A virtual machine is allocated a specific amount of memory, and that is all that it can utilize, even though there might be orders of magnitude more memory available on the physical machine. you can merely reconfigure the amount and the VM will have access to the added capacity, sometimes without even needing a reboot

- **Examining Network Resources in a Virtual Machine**

- Each virtual machine can be configured with one or more network interface cards. The hypervisor supports the creation of a virtual network that connects the virtual NICs to a network that is composed of virtual switches in turn to physical NICs

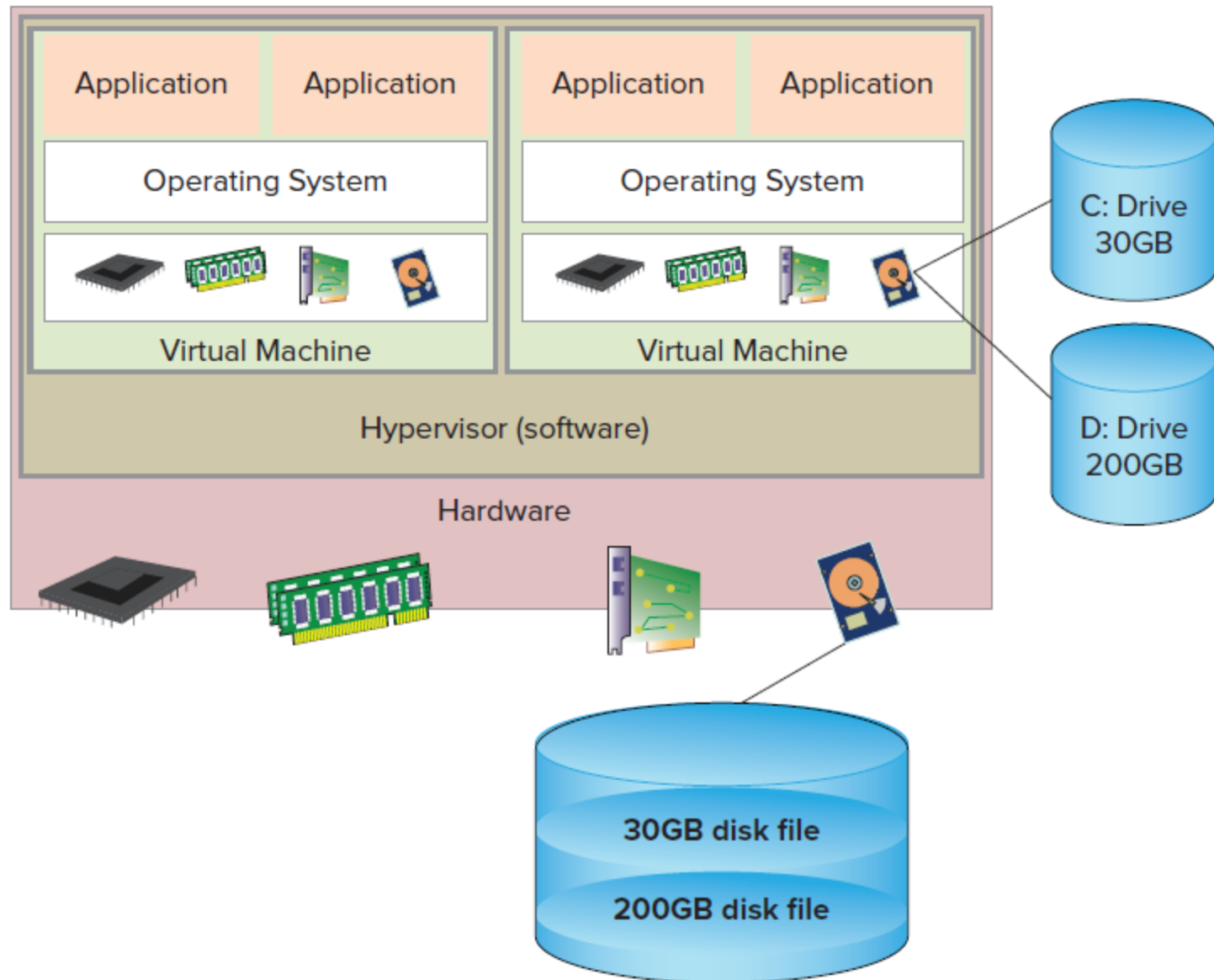
VM-to-VM

communications can occur across a virtual switch and never leave the physical host. If a second VM's virtual NIC connects to a virtual switch, and that switch is not connected to a physical NIC, the only way to communicate with that VM is through the first VM,



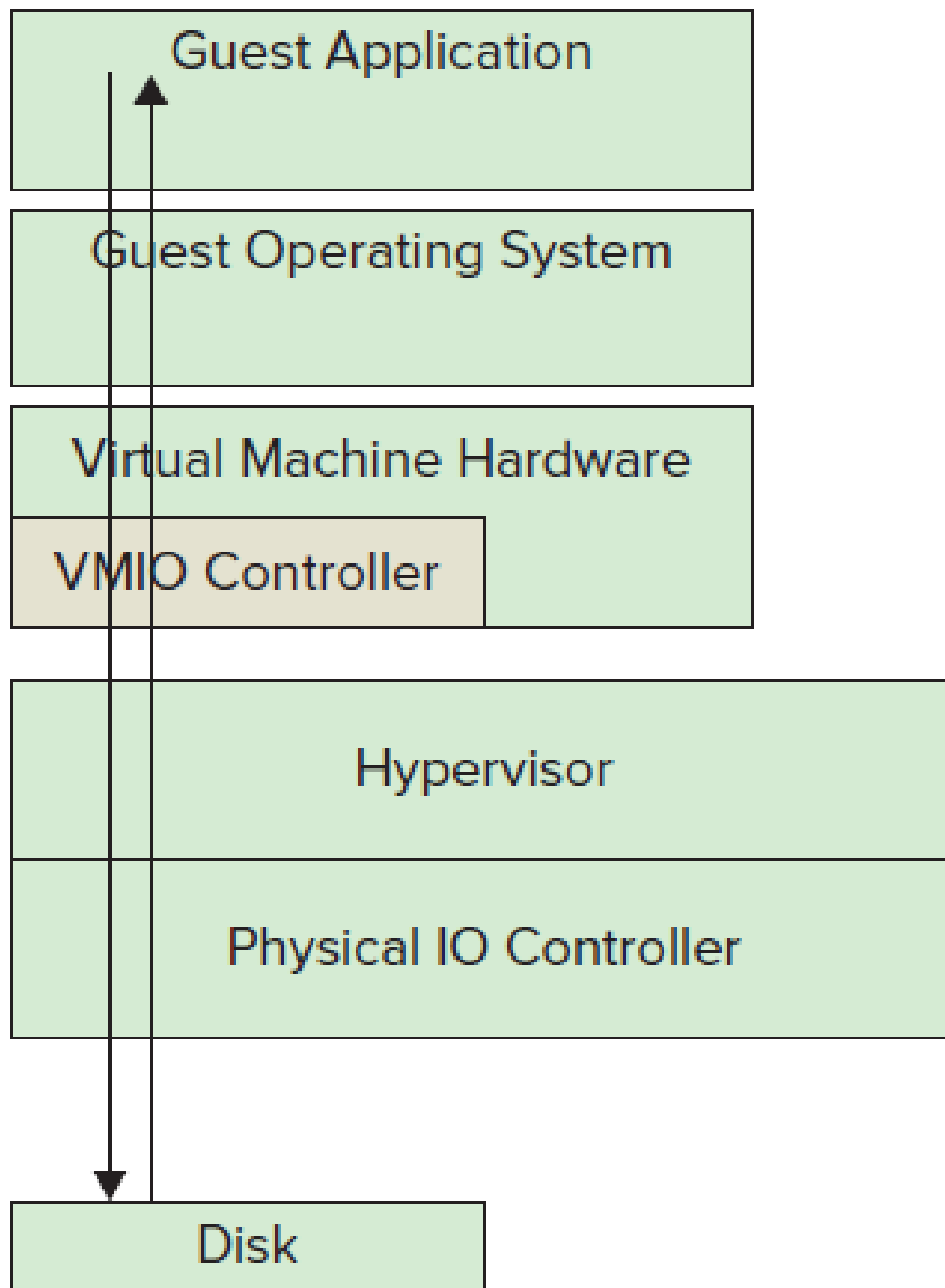
- **Examining Storage in a Virtual Machine**

- Virtual servers need storage to work with. those “drives” are merely carved out regions of disk space on a shared storage device, and the hypervisor manages the presentation to the VM
- As a virtual machine talks to a virtual SCSI disk adapter, the hypervisor passes data blocks to and from the physical storage. That actual connection, from the host to the storage, whether it is local storage on the host or on a storage area network (SAN), is abstracted from the virtual machines.



• **Understanding How a Virtual Machine Works**

- The hypervisor becomes the transporter and regulator of resources to and from the virtual guests it supports. It achieves this capability by fooling the guest operating system into believing that the hypervisor is actually the hardware
- When a program needs some data from a file on a disk, it makes a request through a program language command, such as an `fgets()` in C, which gets passed through to the operating system.
- The operating system has file system information available to it and passes the request on to the correct device manager, which then works with the physical disk I/O controller and storage device to retrieve the proper data.
- The data comes back through the I/O controller and device driver where the operating system returns the data to the requesting program.



- **Working with Virtual Machines**

- **Understanding Virtual Machine Clones**

- If you need a new server, you can clone an existing one. The process involves little more than copying the files that make up the existing server. Once that copy exists, the guest operating system only needs some customization in the form of unique system information, such as a system name and IP address, before it can be instantiated.
 - Tools that manage virtual machines have provisions built in to help with the customizations during cloning, which can make the actual effort itself nothing more than a few mouse clicks

- **Understanding Templates**

- A template is a mold, a preconfigured, preloaded virtual machine that is used to stamp out copies of a commonly used server
 - The difference between a template and a clone is that the clone is running and a template is not

- **Understanding Snapshots**

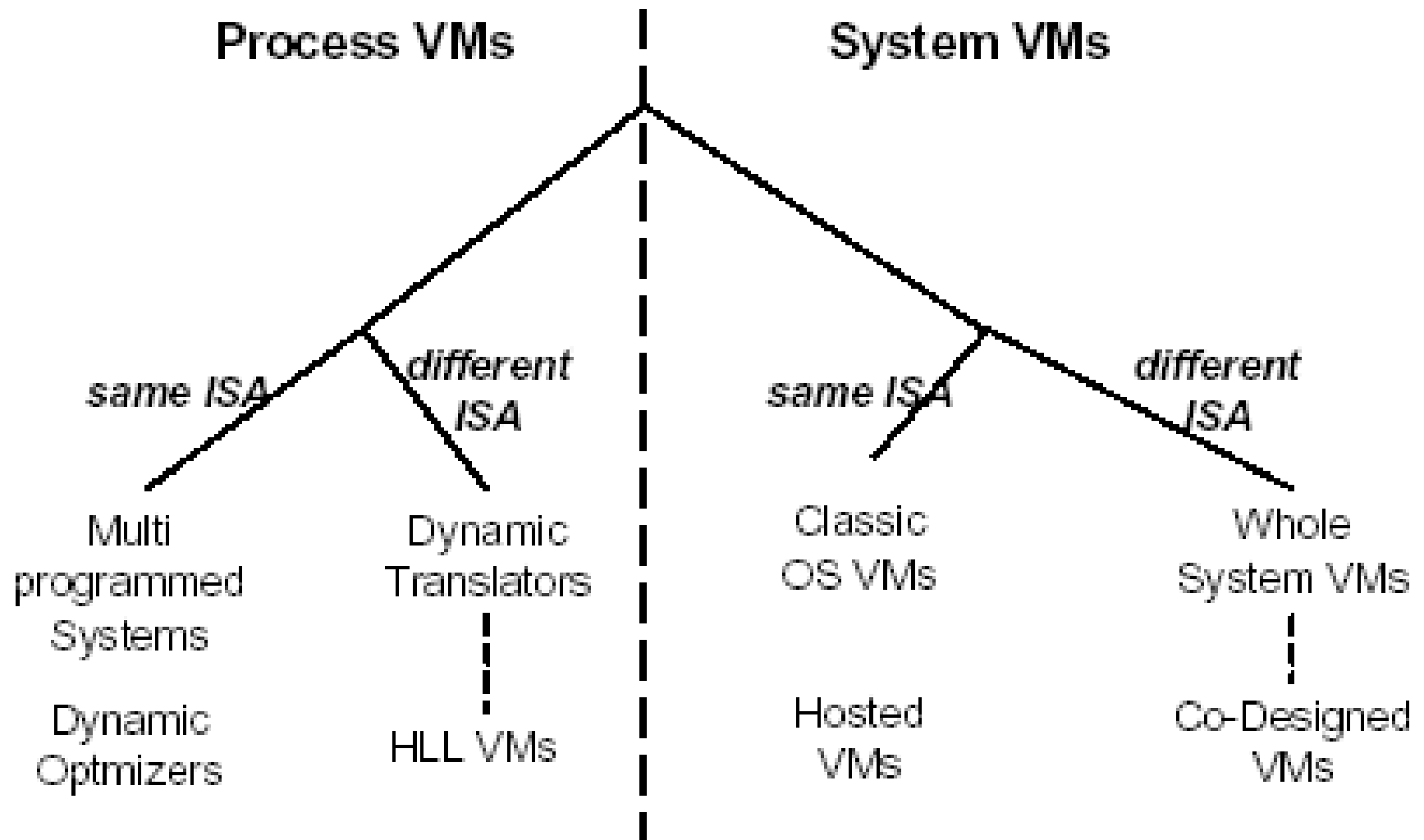
- Snapshot, a capturing of a VM's state at a particular point in time
 - unrolling all of the changes that have been made since that time.
 - A snapshot preserves the state of a VM, its data, and its hardware configuration. Once you snapshot a VM, changes that are made no longer go to the virtual machine. They go instead to a delta disk, sometimes called a child disk. This delta disk accumulates all changes until one of two things happens, another snapshot or a consolidation, ending the snapshot process

• Understanding OVF

- Open Virtualization Format (OVF). OVF is a standard, created by an industry-wide group of people representing key vendors in the various areas of virtualization
- create a platform and vendor-neutral format to bundle up virtual machines into one or more files that can be easily transported from one virtualization platform to another
 - Format 1
 - The OVF template creates a number of files that represent the virtual machine, much as the virtual machine itself is composed of a number of files
 - Format 2
 - OVA, which will encapsulate all of the information in a single file.

Taxonomy of Virtualization/ Virtual Machines

Taxonomy of Virtual Machines



- Process VMs
 - Process-level VMs provide user applications with a virtual ABI environment.
- Same ISA
 - **Multiprogramming**
 - combination of the OS call interface and the user instruction set forms the machine that executes a user process
 - each user process is given the illusion of having a complete machine to itself. Each process is given its own address space and is given access to a file structure. The operating system timeshares the hardware and manages underlying resources to make this possible
 - **Same-ISA Binary Optimizers**
 - Most dynamic binary translators not only translate from source to target code but also perform some code optimizations.
 - A dynamic binary optimizer can collect a profile and then use this profile information to optimize the binary code on the fly

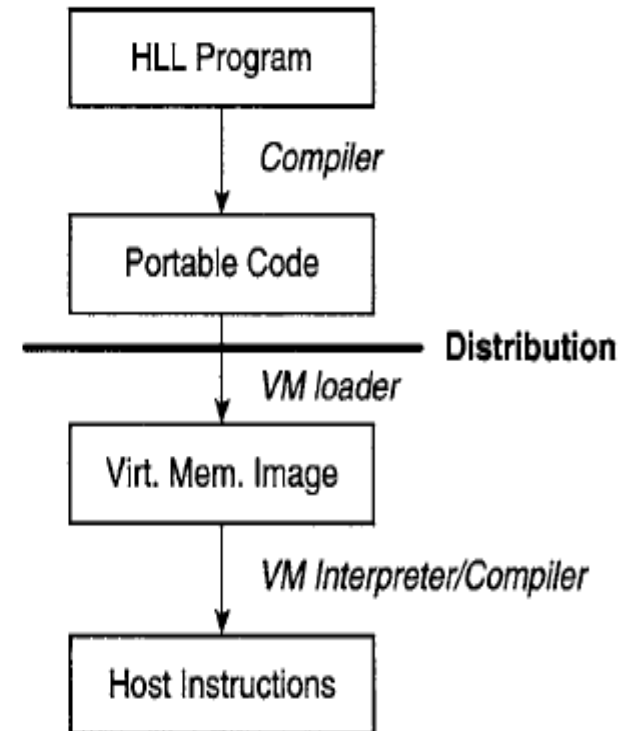
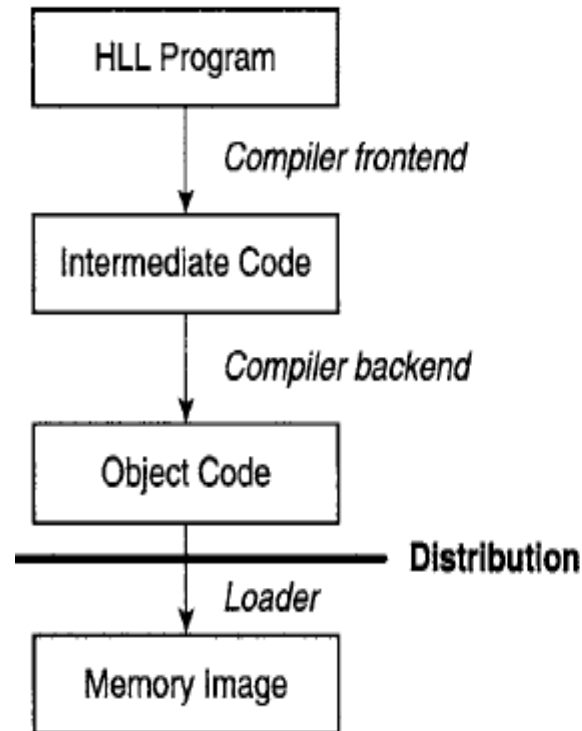
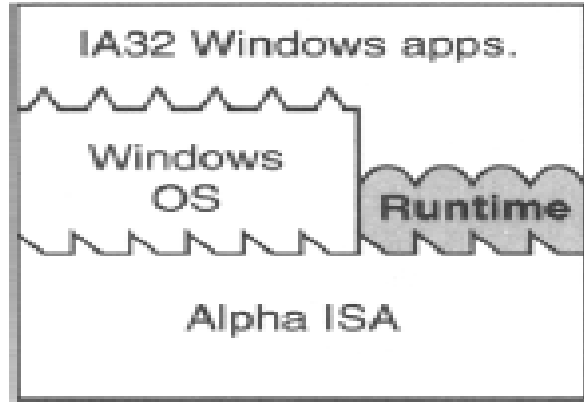
- Different ISA

- **Emulators and Dynamic Binary Translators**

- Support program binaries compiled to a different instruction set than the one executed by the host's hardware, i.e., to emulate one instruction set on hardware designed for another

- **High-Level Language Virtual Machines**

- if one wanted to run IA-32 binaries on a number of hardware platforms currently in use, e.g., SPARC, PowerPC, and MIPS, then an FX!32-like VM would have to be developed for each of them.

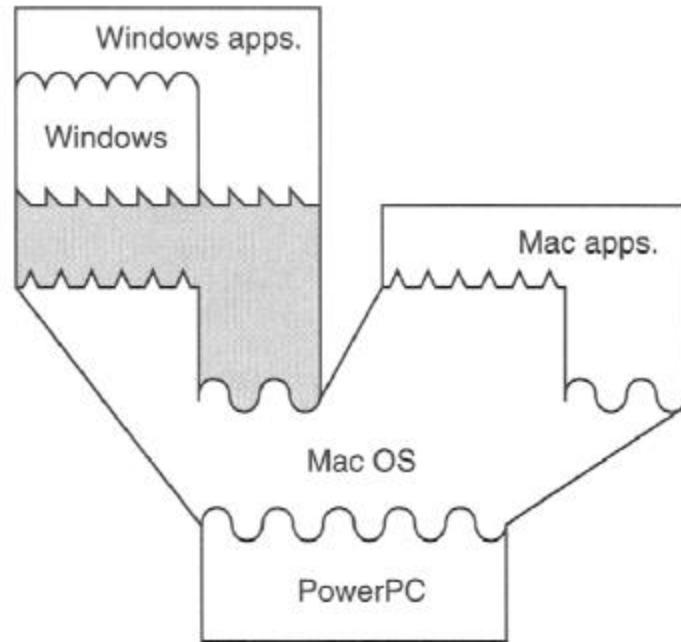


- System VMs
 - System virtual machines provide a complete system environment in which many processes, possibly belonging to multiple users, can coexist
- Same ISA
 - **Classic OS VMs**
 - The VMM is first placed on bare hardware, and virtual machines fit on top. The VMM runs in the most highly privileged mode, while all the guests systems run with lesser privileges. Then in a completely transparent way, the VMM can intercept and implement all the guest OS's actions that interact with hardware resources.
 - Hosted VMs
 - VMM implementation builds virtualizing software on top of an existing host operating system. virtualizing software can rely on the host OS to provide device drivers and other lower-level services; they don't have to be provided by the VMM.

- Different ISA

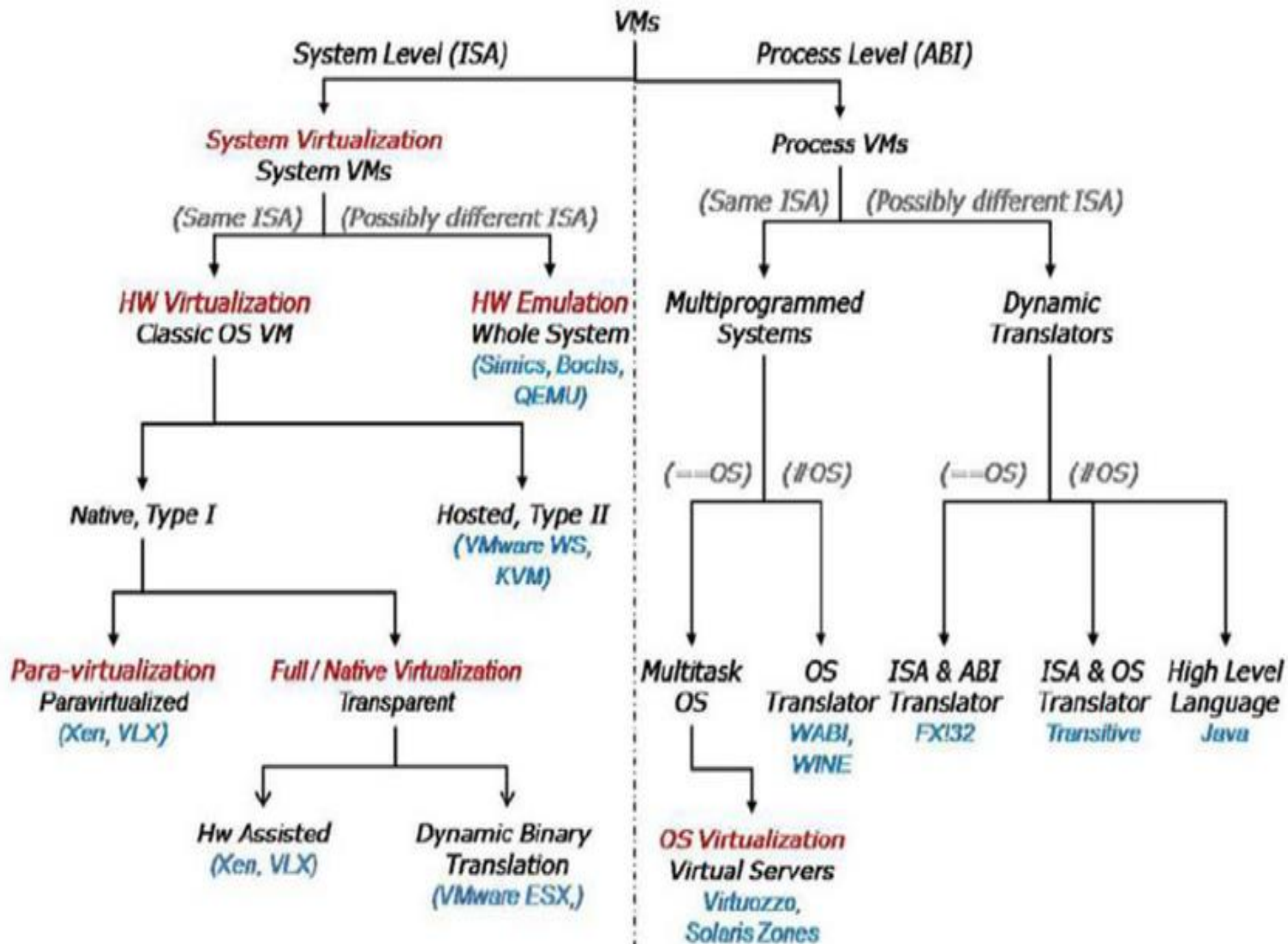
- **Whole System VMs: Emulation**

- system VMs, where a complete software system, both OS and applications, is supported on a host system that runs a different ISA and OS



- **Codesigned Virtual Machines: Hardware Optimization**

- Codesigned VMs are designed to enable innovative ISAs and/or hardware implementations for improved performance, power efficiency, or both. The host's ISA may be completely new, or it may be based on an existing ISA with some new instructions added and/or some instructions deleted.



Questions?