Module 2 Virtualization

Introduction to Virtual Machine (VM) - basics of Virtualization - Types of Virtualizations - Desktop Virtualization – Application Virtualization – Server Virtualization - Storage Virtualization- OS level Virtualization –Virtualization for cloud computing – Software-defined data Center (SDDC).

Introduction to Virtual Machine (VM):

Conventional computer:

1. Single OS image.
2. Rigid Architecture that tightly couples application software to a specific hardware platform. (i.e., software running well on one machine may not be executable on another platform with a different instruction set under a fixed OS.

Virtual machines (VMs) offer new solutions to

1. underutilized resources,
2. application inflexibility,
3. software manageability and
4. Security concerns in existing physical machines.

A traditional computer runs with a host operating system specially tailored for its hardware architecture, as shown in Figure **1(a).** After virtualization, different user applications managed by their own operating systems (guest OS) can run on the same hardware, independent of the host OS. This is often done by adding additional software, called a *virtualization layer* as shown in Figure**1 (b).** This virtualization layer is known as *hypervisor* or *virtual machine monitor* (VMM). The VMs are shown in the upper boxes, where applications run with their own guest OS over the virtualized CPU, memory, and I/O resources.

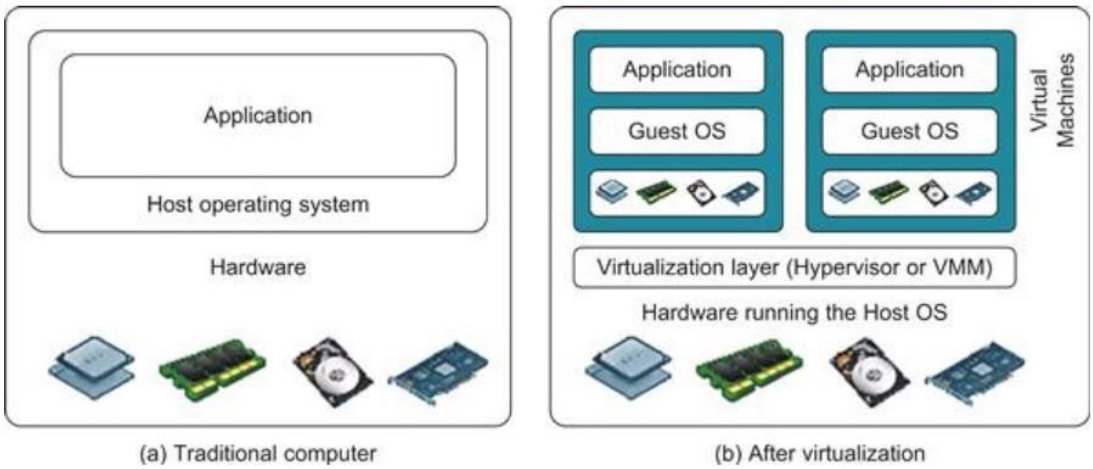


FIGURE 1. The architecture of a computer system before and after virtualization, where VMM stands for virtual machine monitor.

Hypervisor

Defn-1: is known as Virtual Machine Monitor/Manager) is a software that allows multiple OS to share a single Hardware host.

Defn-2: A Software that creates and runs Virtual Machine.

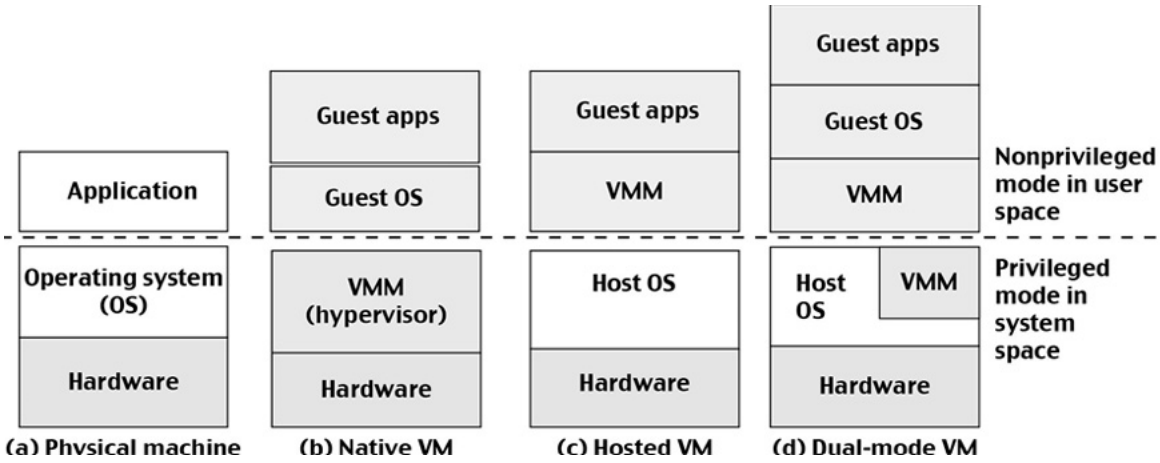
Defn-3: Allows multiple guest OS to run on a single host.

Virtual Machine

Defn-1: is an emulation of a particular computer system.

Defn-2: VM is built with virtual resources managed by a guest OS to run a specific application.

Comparison between Physical Machine, Native VM, Hosted VM, Dual Mode VM



- **Physical Machine:**
    - Equipped with the physical hardware
    - Example: x-86 architecture desktop running its installed Windows OS

- **Virtual Machine:**
    - Can be provisioned for any hardware system.
    - Built with virtual resources managed by a guest OS to run a specific application.
    - Deploys a middleware layer called a virtual machine monitor (VMM) between the VMs and the host platform.
- **Native VM:**
    - Installed with the use of a VMM called a hypervisor in privileged mode.

• Guest OS could be a Linux system and the hypervisor is the XEN system.

• Also called bare-metal VM, because the hypervisor handles the bare hardware (CPU, memory, and I/O) directly.

- **Hosted VM:**

• VMM runs in non-privileged mode.

• The host OS need not be modified.

• Dual mode VM:

• Part of the VMM runs at the user level and another part runs at the supervisor level.

• The host OS may have to be modified to some extent.

VM Primitive Operations

• The VMM provides the VM abstraction to the guest OS.

• With full virtualization, the VMM exports a VM abstraction identical to the physical machine so that a standard OS such as Windows 2000 or Linux can run just as it would on the physical hardware.

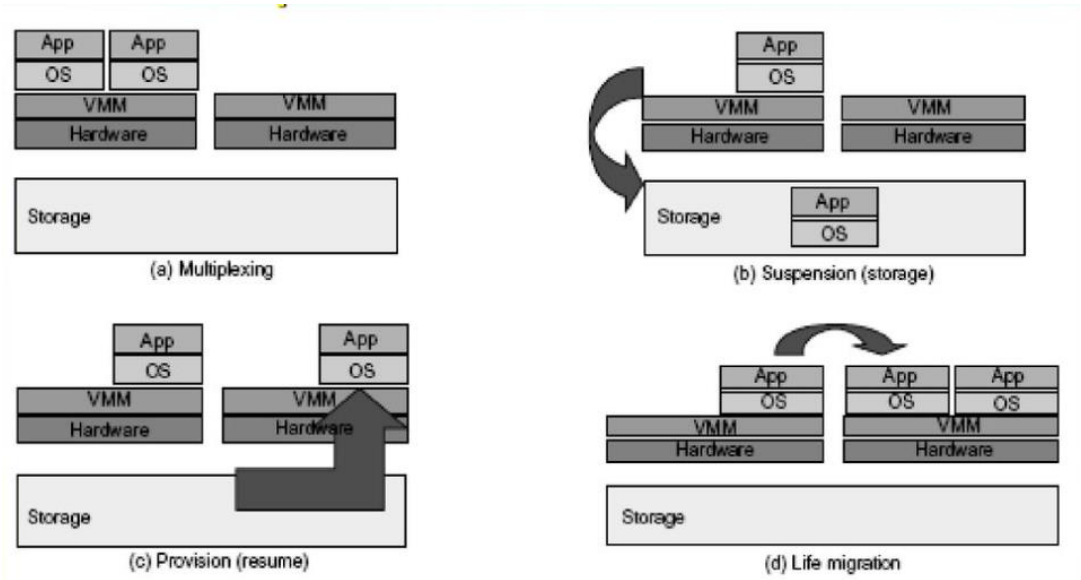
•Low-level operations of VMM are

a. VMs can be **multiplexed** between hardware machines,

b. VM can be suspended and stored in stable storage

c. suspended VM can be resumed or provisioned to a new hardware platform

d. VM can be migrated from one hardware platform to another



Uses of Virtual Machines (VMs):

Cloud Computing:

VMs enable the virtualization of physical hardware, forming the backbone of cloud services. They allow scalable distribution of virtual resources over the internet, powering platforms like Dropbox, Salesforce, and Google Drive.

Software Development and Testing:

By creating isolated environments, VMs let developers test software without affecting the host system or other applications, ensuring safe and efficient development workflows.

Malware Analysis:

VMs provide a secure space for examining malicious software, protecting the main system and network from potential threats.

Disaster Recovery:

Replicating systems on VMs ensures data and functionality can be quickly restored in case of failure, theft, or damage. For example, iCloud backups allow users to restore their data onto new devices seamlessly.

Legacy Software Support:

VMs can emulate older operating systems, making it possible to run outdated applications on modern hardware.

Advantages and disadvantages of virtual machines

Advantages of VMs	**Disadvantages of VMs**
Portability: VMs allow users to move systems to other computing environments easily.	Infected VMs. It can be risky to create VMs from weak host hardware. An improperly structured host system may spread its OS bugs to VMs.
Speed: Creating a VM is much faster than installing a new OS on a physical server. VMs can also be cloned, OS included.	Server sprawl. The ability to create virtual machines can quickly lead to a crowded network. It's best to monitor the creation of VMs to preserve computational resources.
Security:VMs help provide an extra layer of security because they can be scanned for malware. They also enable users to take snapshots of their current states. If an issue arises, users can review those snapshots to trace it and restore the VM to a previous version.	Complexity. System failures can be challenging to pinpoint in infrastructure with multiple local area networks (LANs).

Basics of Virtualization

- Virtualization is the process of creating a software-based, or "virtual," version of a computer.
- It allocates dedicated amounts of CPU, memory, and storage "borrowed" from a physical host computer or a remote server in a cloud provider's datacenter.
- A virtual machine is a computer file, typically referred to as an *image*, that functions like an actual computer.
- It runs as a separate computing environment within a window on the host system.
- Often used to run a different operating system or act as the user's entire computing experience (common on work computers).
- A VM is isolated from the host system, ensuring that the software inside the VM cannot interfere with the primary operating system of the host computer.

LEVELS OF VIRTUALIZATION IMPLEMENTATION

The main function of the software layer for virtualization is to virtualize the physical hardware of a host machine into virtual resources to be used by the VMs, exclusively. This can be implemented at various operational levels, as we will discuss shortly. The virtualization software creates the abstraction of VMs by interposing a virtualization layer at various levels of a computer system. Common virtualization layers include the *instruction set architecture (ISA)* level, hardware level, operating system level, library support level, and application level (see Figure 2).

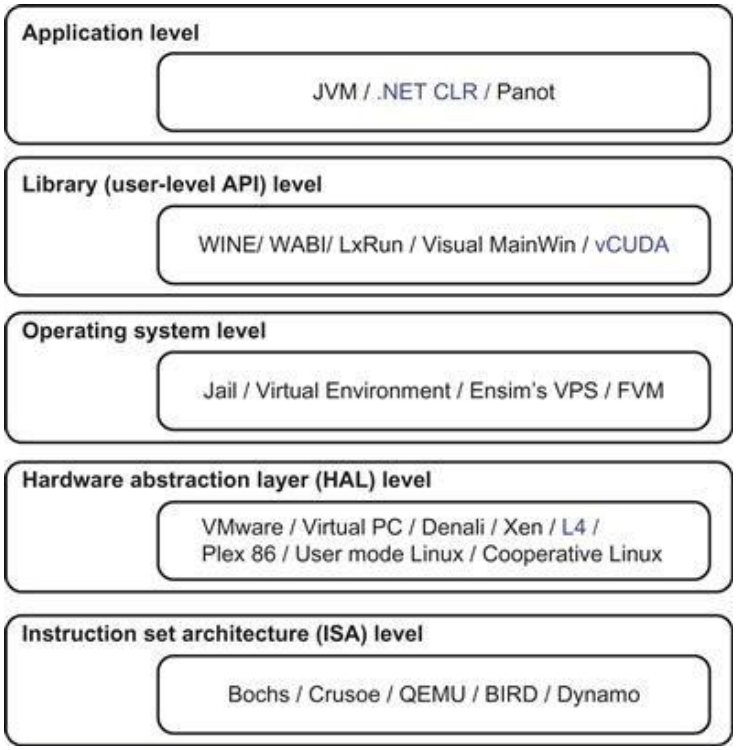


FIGURE 2 Virtualization ranging from hardware to applications in five abstraction levels.

Instruction Set Architecture Level

At the ISA level, virtualization is performed by emulating a given ISA by the ISA of the host machine. For example, MIPS binary code can run on an x86-based host machine with the help of ISA emulation. With this approach, it is possible to run a large amount of legacy binary code written for various processors on any given new hardware host machine. Instruction set emulation leads to virtual ISAs created on any hardware machine.

The basic emulation method is through *code interpretation*. An interpreter program interprets the source instructions to target instructions one by one. One source instruction may require tens or hundreds of native target instructions to perform its function. Obviously, this process is relatively slow. For better performance, *dynamic binary translation* is desired. This approach translates basic blocks of dynamic source instructions to target instructions. The basic blocks can also be extended to program traces or super blocks to increase translation efficiency. Instruction set emulation requires binary translation and optimization. A *virtual instruction set architecture (V-ISA)* thus requires adding a processor-specific software translation layer to the compiler.

Hardware Abstraction Level

Hardware-level virtualization is performed right on top of the bare hardware. On the one hand, this approach generates a virtual hardware environment for a VM. On the other hand, the process manages the underlying hardware through virtualization. The idea is to virtualize a computer's resources, such as its processors, memory, and I/O devices. The intention is to upgrade the hardware utilization rate by multiple users concurrently. The idea was implemented in the IBM VM/370 in the 1960s. More recently, the Xen hypervisor has been applied to virtualize x86-based machines to run Linux or other guest OS applications.

Operating System Level

This refers to an abstraction layer between traditional OS and user applications. OS-level virtualization creates isolated *containers* on a single physical server and the OS instances to utilize the hardware and software in data centers. The containers behave like real servers. OS-level virtualization is commonly used in creating virtual hosting environments to allocate hardware resources among a large number of mutually distrusting users.

Library Support Level

Most applications use APIs exported by user-level libraries rather than using lengthy system calls by the OS. Since most systems provide well-documented APIs, such an interface becomes another candidate for virtualization. Virtualization with library interfaces is possible by controlling the communication link between applications and the rest of a system through API hooks. The software tool WINE has implemented this approach to support Windows applications on top of UNIX hosts. Another example is the vCUDA which allows applications executing within VMs to leverage GPU hardware acceleration.

User-Application Level

Virtualization at the application level virtualizes an application as a VM. On a traditional OS, an application often runs as a process. Therefore, *application-level virtualization* is also known

as *process-level virtualization*. The most popular approach is to deploy *high level language (HLL)* VMs. In this scenario, the virtualization layer sits as an application program on top of the operating system, and the layer exports an abstraction of a VM that can run programs written and compiled to a particular abstract machine definition. Any program written in the HLL and compiled for this VM will be able to run on it. The Microsoft .NET CLR and *Java Virtual Machine (JVM)* are two good examples of this class of VM.

Other forms of application-level virtualization are known as *application isolation*, *application sandboxing*, or *application streaming*. The process involves wrapping the application in a layer that is isolated from the host OS and other applications. The result is an application that is much easier to distribute and remove from user workstations. An example is the LANDesk application virtualization platform which deploys software applications as self-contained, executable files in an isolated environment without requiring installation, system modifications, or elevated security privileges.

Relative Merits of Different Approaches

Table1 compares the relative merits of implementing virtualization at various levels. The column headings correspond to four technical merits. "Higher Performance" and "Application Flexibility" are self-explanatory. "Implementation Complexity" implies the cost to implement that particular virtualization level. "Application Isolation" refers to the effort required to isolate resources committed to different VMs. Each row corresponds to a particular level of virtualization.

Table 1.Relative Merits of Virtualization at Various Levels (More "X"'s Means Higher Merit, with a Maximum of 5 X's)

Level of Implementation	Higher Performance	Application Flexibility	Implementation Complexity	Application Isolation
ISA	X	XXXXX	XXX	XXX
Hardware-level virtualization	XXXXX	XXX	XXXXX	XXXX
OS-level virtualization	XXXXX	XX	XXX	XX
Runtime library support	XXX	XX	XX	XX
User application level	XX	XX	XXXXX	XXXXX

The number of X's in the table cells reflects the advantage points of each implementation level. Five X's implies the best case and one X implies the worst case. Overall, hardware and OS support will yield the highest performance. However, the hardware and application levels are also the most expensive to implement. User isolation is the most difficult to achieve. ISA implementation offers the best application flexibility.

Benefits of Virtualization

- More flexible and efficient allocation of resources.
- Enhance development productivity.
- It lowers the cost of IT infrastructure.
- Remote access and rapid scalability.
- High availability and disaster recovery.

- Pay peruse of the IT infrastructure on demand.
- Enables running multiple operating systems.

Drawback of Virtualization

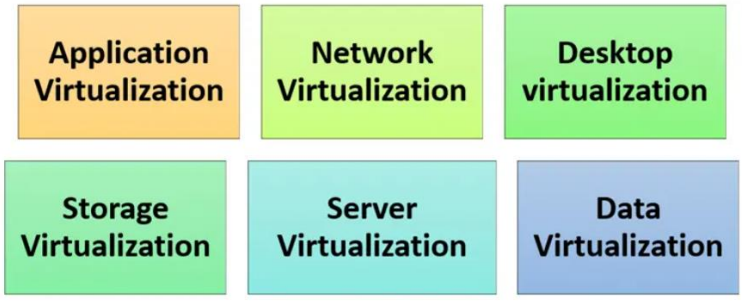
- **High Initial Investment:** Clouds have a very high initial investment, but it is also true that it will help in reducing the cost of companies.
- **Learning New Infrastructure:** As the companies shifted from Servers to Cloud, it requires highly skilled staff who have skills to work with the cloud easily, and for this, you have to hire new staff or provide training to current staff.
- **Risk of Data:** Hosting data on third-party resources can lead to putting the data at risk, it has the chance of getting attacked by any hacker or cracker very easily.

Characteristics of Virtualization

- **Increased Security:** The ability to control the execution of a guest program in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment. All the operations of the guest programs are generally performed against the virtual machine, which then translates and applies them to the host programs.
- **Managed Execution:** In particular, sharing, aggregation, emulation, and isolation are the most relevant features.
- **Sharing:** Virtualization allows the creation of a separate computing environment within the same host.
- **Aggregation:** It is possible to share physical resources among several guests, but virtualization also allows aggregation, which is the opposite process.

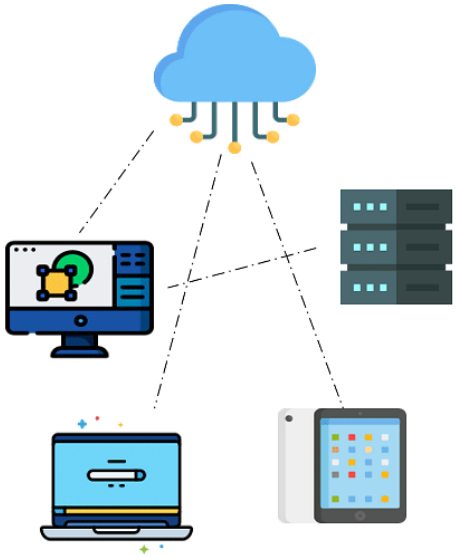
Types of Virtualization

1. Application Virtualization
2. Network Virtualization
3. Desktop Virtualization
4. Storage Virtualization
5. Server Virtualization
6. Data virtualization

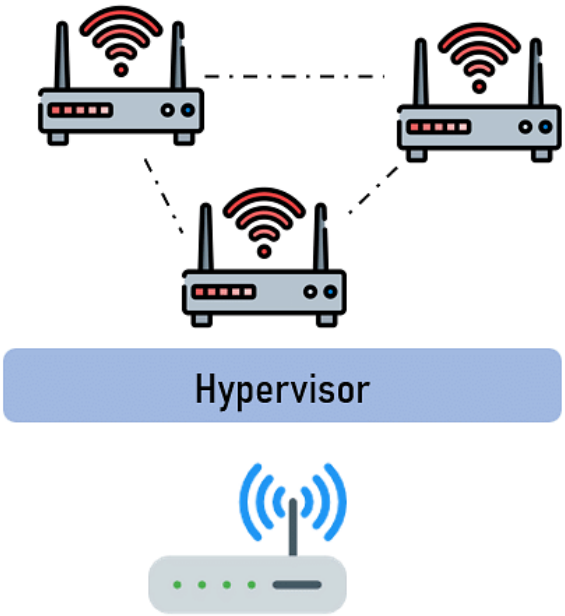


Types of Virtualizations

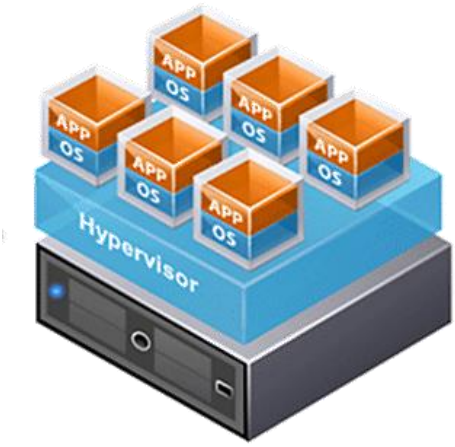
a) **Application Virtualization**: Enables users to access applications remotely from a server that stores all personal data and application characteristics. The application can run locally via the internet, useful for running multiple software versions. Examples include hosted and packaged applications.



b) **Network Virtualization**: Runs multiple virtual networks with separate control and data planes over a single physical network. It supports virtual networks, switches, routers, firewalls, VPNs, and workload security, ensuring efficient provisioning and management.



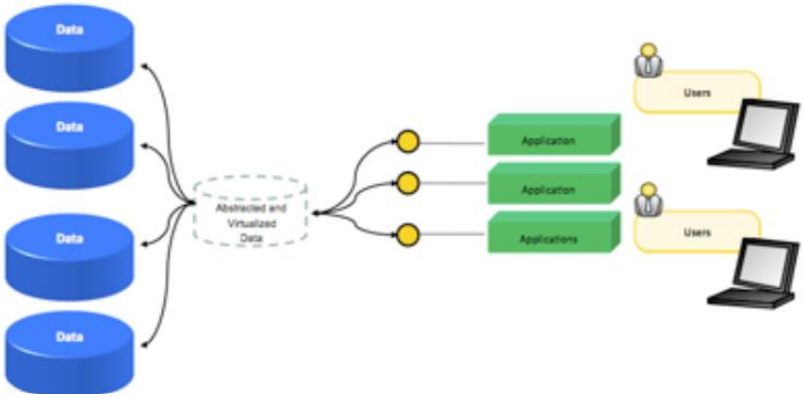
c) **Desktop Virtualization**: Stores a user's OS on a server, allowing access from any device or location. This enables user mobility, portability, and easier software management for updates and patches.



d) **Storage Virtualization**: Combines multiple storage sources into a single virtual repository, ensuring smooth operation, consistent performance, and advanced functionality despite underlying hardware differences.

e) **Server Virtualization**: Divides a physical server into multiple virtual servers with independent OS operations. This improves performance, reduces energy and infrastructure costs, and supports virtual migration.

f) **Data Virtualization**: Aggregates data from various sources into a single logical view accessible remotely via cloud services, without needing details about data storage or formatting. Major providers include Oracle, IBM, and AtScale.



Desktop virtualization

Desktop virtualization is an innovative technology that detaches the desktop environment, including the operating system, applications, and data, from the physical machine. When the tools are detached from the machine itself, it allows for a highly flexible and accessible computing system where the user's desktop is hosted on a server and can be accessed from anywhere.

In the realm of computing, the concept of desktop virtualization serves as a bridge between the traditional, physical constraints of hardware and the limitless potential of digital workspaces. It mirrors the shift in our perception and use of computers and empowers users to access their personal desktop space remotely, providing flexibility and mobility unheard of in the traditional computing model.

Desktop virtualization represents a significant leap towards more agile, resilient, and user-centric computing models, breaking down the barriers imposed by traditional IT infrastructure.

How does it work?

At its core, desktop virtualization operates by hosting a desktop operating system on a centralized server. This setup allows multiple users to access their own virtualized desktop instances simultaneously. When a user logs in, they're connected to their desktop instance running on the server. This connection can be made through various devices—be it a traditional PC, a thin client, a tablet, or a smartphone—offering a seamless computing experience regardless of the hardware used.

This versatile solution works in two primary ways: local and remote.

Local desktop virtualization

With local desktop virtualization, the computer's operating system is run directly on a client device, leveraging the local system resources. This approach is particularly suited for those who do not require constant network connection and whose computing needs fit within the local system capacity. When processing is done locally though, local desktop virtualization doesn’t allow for sharing virtual machines (VMs) or external resources across a network including mobile devices and thin clients.

Remote desktop virtualization
On the other hand, remote desktop virtualization shines in server-based environments. It enables users to operate systems and applications housed within the secure confines of a datacentre while engaging with them on personal devices like laptops or smartphones. This setup offers IT teams the advantage of centralized management and allows organizations to stretch their hardware investments by providing remote access to pooled computing power.

Types of desktop virtualization

There are two types of desktop virtualization, hosted and client.

Hosted virtualization
Hosted desktop virtualization involves hosting desktop environments on a central server or in the cloud. This category can be broken down into several types:

- **Virtual desktop infrastructure (VDI)** makes desktops and applications an on-demand service, allowing access anytime and anywhere. With virtual desktop infrastructure, each user receives a dedicated desktop instance on the server, which frees them up to use any device to access their instance. This method offers a high degree of personalization and performance but requires significant server resources.

- Remote desktop services (RDS) enables multiple users to access a shared desktop and applications from a remote server. With RDI, multiple users share a single operating system instance, optimizing resource use but offering less personalization.

- Desktop-as-a-Service (DaaS) delivers hosted desktop services from a third party. With desktop-as-a-service, organizations can give employees anytime-anywhere access to personalized desktops from virtually any device. This cloud-based service shifts the burden of managing the backend responsibilities of data storage, backup, security, and upgrades to the provider. DaaS offers scalability and flexibility as hybrid environments are increasingly common, making it an attractive option for small to medium-sized businesses.

Client virtualization
The other type of desktop virtualization, client virtualization, brings a different approach, focusing on running the virtualization technology directly on the user's device. This category can be separated into two types:

- **Presentation virtualization** separates the application layer from the graphical user interface, displaying the application on the user's device while it runs on a server. It can support resource efficiency and management and is useful in settings where many users need to access a standardized set of applications and where the central control and management of these applications are critical.

- **Application virtualization** separates an application from the underlying computer hardware it is stored on. With application visualization, applications are allowed to run without being directly installed on the operating system. This method simplifies application deployment and management and is useful in settings where apps need to be accessed remotely on varied devices.

Benefits of desktop virtualization:

Desktop virtualization offers numerous benefits especially as the nature of work environments and data management continues to evolve and change:

- **Enhanced security** - Storing business critical data within a datacenter enhances security because it eliminates the risks associated with data that is stored on local devices. With data and applications stored in secure datacenters, the risk of data theft from lost or stolen devices is minimized. Furthermore, desktop virtualization allows for better control over access to sensitive information, as data never leaves the datacenter and can be quickly wiped from devices if an employee leaves the company.

- **Simplified management and workflows** - IT departments can manage and update desktops and permissions centrally, reducing the complexity and cost of desktop management. Desktop personalization eliminates the need for manually setting new desktops for each user, since IT can easily deploy a packaged virtual desktop to the user's device. The process for updating across devices is much less involved for IT teams when application and operating systems data is stored in centralized locations, instead of on individual users' machines.

- **Cost savings and resource management** - Organizations can save on hardware costs by extending the lifecycle of older devices and reducing the need for expensive client hardware and upgrades. When users' machines no longer need to do all the computing internally, companies can save money on device capabilities with more affordable machines. From a people ops perspective, centralizing desktop management can significantly reduce IT overhead and bolster revenue margins.

- **Flexibility and streamlined experience** - Users can access their desktops and applications from any device, anywhere, at any time. The ability to access your personalized computer from anywhere and using any device, one of the most tangible benefits for end-users, is a game-changer for remote work, education, and even personal computing. This flexibility improves employee experience and affords new possibilities for how and where people can work.

Challenges come with employing desktop virtualization:

Despite its many benefits, desktop virtualization also presents a few challenges.

- **High touch engagement** - The infrastructure required for desktop virtualization can be complex to set up and manage. The initial setup and ongoing management of desktop virtualization requires a deep understanding of both the technology and the specific needs of the organization.

- **Performance issues** - Ensuring high performance and low latency can be difficult, especially over wide-area networks. Graphics-intensive applications or usage in low-bandwidth environments can frustrate users and hamper productivity.

- **Upfront cost** - The cost of implementing a desktop virtualization solution can also be a barrier. While there are long-term savings to be had, the upfront investment in server hardware, software licensing fees, and network infrastructure can be significant.

- **Ongoing complexity** - Navigating the complex licensing agreements for virtualized desktops can be a challenge for some IT departments.

Use cases

Desktop virtualization is highly versatile, catering to several use cases:

- **Remote work** - Facilitates secure and efficient remote access to work environments. In today's world of hybrid and remote roles, it allows employees to access their work environment securely from anywhere.

- **Education** - Provides students access to learning resources from any device. Desktop virtualization enables the virtualization of computer labs, providing students with access to specialized software without the need for high-end personal computers.

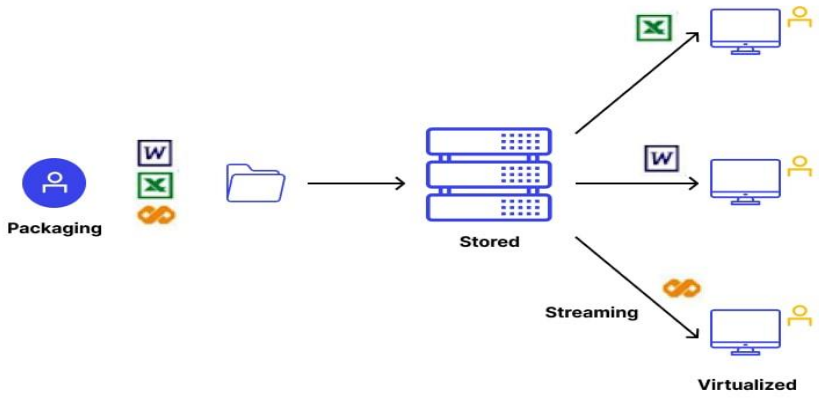
- **Healthcare** – Ensures critical information is always available for key staff at healthcare institutions. Doctors and staff can access patient records and applications securely and efficiently, from any location.

Application virtualization:

- It is a technique used to trick conventional software into thinking it directly interacts with an OS's capabilities when it does not. A virtualization barrier must be installed between the program and the OS for this trick to work.
- This top-notch barrier or platform has to be able to theoretically run an app's components without affecting the OS below it. By seamlessly redirecting documents and database log modifications to a single piece of software, this procedure takes over a component of the execution environment traditionally provided by the OS.
- The program may now exist alongside previously incompatible programs since its processes are consolidated into a single file rather than spreading out over the OS.

Working of Application Virtualization:

- The method is effective for the type of software that enables users to access an app from a device other than the one on which it should be properly deployed.
- Using this technique, the program behaves and communicates just as it would if it were downloaded directly on the user's computer. The client will have a satisfying experience because of this. Also, the user does not need to deploy the program on the actual device in order to utilize it.
- The IT manager often distributes remote apps to a user's computer or other wireless connections from a centralized computer in the association's data storage. Now the customer may utilize and view the program center. The program is then sent to users as though it were natively deployed on their machine using application virtualization software. The server will then carry out the customer's commands.



This diagram illustrates the process of **application virtualization**, which allows applications to run on end-user devices without being directly installed on them. Here's the explanation:

1. **Packaging**:
    - Applications like Microsoft Word, Excel, or other software are prepared (packaged) in a way that allows them to be deployed remotely. This step involves creating a virtualized version of the application, ensuring it can run independently of the underlying operating system or hardware.
2. **Storage**:
    - The packaged applications are stored centrally on servers. These servers act as a repository, managing and distributing the virtualized applications to users on demand. The applications are not physically installed on the user's devices but remain on the server.
3. **Streaming**:
    - When users request an application, it is streamed from the server to their device. This allows the application to run locally on the user's device while still being centrally managed. Streaming ensures that only the necessary parts of the application are sent, reducing load times.
4. **Virtualized Usage**:
    - Users access the virtualized applications on their devices. The applications function as if they were locally installed, but in reality, they are being delivered and executed through the virtualized environment. This allows multiple users to use different versions of the same software without conflicts.

Benefits of Application Virtualization :

Centralized control: Those Cloud-based applications can be run virtually, eliminating the need for updates first installed on end-users' gadgets.

Enhanced security: This is because the applications are run within the secure data centers provided within the Cloud. Access controls and security measures can be implemented on the application level.

Enhanced flexibility: Employees can access various applications virtually anywhere with an internet connection, which enhances the implementation of 'work from home' policies or 'bring your own device' policies.

Cost savings: The traditional approach for hardware and software licenses on end-user devices is unnecessary when using cloud-based application virtualization. Users can use inexpensive thin clients or their devices to reach applications.

Scalability: Cloud computing offers abundant resources to adjust application virtualization according to demand, guaranteeing steady performance for users.

Decreased software conflicts: Virtualizing applications prevents conflicts between various software versions or dependencies, as each application operates in a separate isolated environment.

Disadvantages of application virtualization

The advantages of virtualized environments are numerous and include some of the following, which are related to the proliferation of mobile and mixed working environments:

- **Simple Installation:** The configuration process is straightforward. And once it completes, you can easily virtualize an app to execute in several endpoints. It is no longer recommended to install the program on every terminal.
- **Simple deployment:** The apps are also simple to install for customers or suppliers. The deployment of these programs is much simpler if you only provide them with the executables that have already been set up.
- **Programs are easy to remove:** All you have to do is eliminate virtualized apps. There is no need to remove the software from each machine.
- **Easy firmware upgrades:** Instead of updating each desktop separately, you can upgrade the virtual programs once from a centralized location.
- **Improved Support:** Help desk employees may observe and address problems with the functioning of virtualized apps from a centralized location if there are any.
- **Liberation from the OSs:** Virtualized programs may be utilized on any terminal, whether it runs Microsoft, iOS, or Android because they are separate from the host platform.

Use Cases

1. **Flexibility in Application**

For simplicity of use, corporate apps ought to be available on every companion smartphone. By enabling the delivery of programs to any interface, it provides application flexibility.

2. **Migrations Made Simple**

Because virtualization technology isolates programs/apps from the underlying infrastructure, you don't need to perform expensive conversions from one sort of OS to the other.

3. **Cost management**

Buying pricey PCs for each one of your workers or customers might become outrageously costly if you have got a large workforce. In this circumstance, app virtualization comes to your aid by enabling you to deploy vital apps to every terminal.

4. **Using Internal Apps**

The deployment of internal programs that programmers often update is also possible through virtualization. It enables remote and rapid upgrades, installations, and distribution of these programs. For companies that employ apps, application virtualization is extremely vital.

5. **Remote Access Security Features**

Professionals may securely access essential programs from wherever thanks to virtualized environments. It is helpful in work-from-home situations since it offers reliability and safety.

Server Virtualization

Server Virtualization is the process of dividing a physical server into several virtual servers, called **virtual private servers**. Each virtual private server can run independently.

The concept of Server Virtualization widely used in the IT infrastructure to minimizes the costs by increasing the utilization of existing resources.

Types of Server Virtualization

1. Hypervisor

In the Server Virtualization, Hypervisor plays an important role. It is a layer between the operating system (OS) and hardware. There are two types of hypervisors.

- Type 1 hypervisor (also known as bare metal or native hypervisors)
- Type 2 hypervisor (also known as hosted or Embedded hypervisors)

The hypervisor is mainly used to perform various tasks such as allocate physical hardware resources (CPU, RAM, etc.) to several smaller independent virtual machines, called "**guest**" on the host machine.

2. Full Virtualization

Full Virtualization uses a **hypervisor** to directly communicate with the CPU and physical server. It provides the best isolation and security mechanism to the virtual machines.

The biggest disadvantage of using hypervisor in full virtualization is that a hypervisor has its own processing needs, so it can slow down the application and server performance.

VMWare ESX server is the best example of full virtualization.

3. Para Virtualization

Para Virtualization is quite similar to the Full Virtualization. The advantage of using this virtualization is that it is **easier to use**, **Enhanced performance**, and **does not require emulation overhead**. Xen primarily and UML use the Para Virtualization.

The difference between full and pare virtualization is that, in para virtualization hypervisor does not need too much processing power to manage the OS.

4. Operating System Virtualization

Operating system virtualization is also called as system-lever virtualization. It is a **server virtualization technology** that divides one operating system into multiple isolated user-space called **virtual environments**. The biggest advantage of using server visualization is that it reduces the use of physical space, so it will save money.

Linux OS Virtualization and **Windows OS Virtualization** are the types of Operating System virtualization.

FreeVPS, **OpenVZ**, and **Linux Vserver** are some examples of System-Level Virtualization.

Note: OS-Level Virtualization never uses a hypervisor.

5. Hardware Assisted Virtualization

Hardware Assisted Virtualization was presented by **AMD and Intel**. It is also known as **Hardware virtualization**, **AMD virtualization**, and **Intel virtualization**. It is designed to increase the performance of the processor. The advantage of using Hardware Assisted Virtualization is that it requires less hypervisor overhead.

6. Kernel-Level Virtualization

Kernel-level virtualization is one of the most important types of server virtualization. It is an **open-source virtualization** which uses the Linux kernel as a hypervisor. The advantage of using kernel virtualization is that it does not require any special administrative software and has very less overhead.

User Mode Linux (UML) and **Kernel-based virtual machine** are some examples of kernel virtualization.

Advantages of Server Virtualization

There are the following advantages of Server Virtualization -

1. Independent Restart

In Server Virtualization, each server can be restart independently and does not affect the working of other virtual servers.

2. Low Cost

Server Virtualization can divide a single server into multiple virtual private servers, so it reduces the cost of hardware components.

3. Disaster Recovery<

Disaster Recovery is one of the best advantages of Server Virtualization. In Server Virtualization, data can easily and quickly move from one server to another and these data can be stored and retrieved from anywhere.

4. Faster deployment of resources

Server virtualization allows us to deploy our resources in a simpler and faster way.

5. Security

It allows uses to store their sensitive data inside the data centers.

Disadvantages of Server Virtualization

1. The biggest disadvantage of server virtualization is that when the server goes offline, all the websites that are hosted by the server will also go down.
2. There is no way to measure the performance of virtualized environments.
3. It requires a huge amount of RAM consumption.
4. It is difficult to set up and maintain.
5. Some core applications and databases are not supported virtualization.
6. It requires extra hardware resources.

Uses of Server Virtualization

A list of uses of server virtualization is given below -

- Server Virtualization is used in the testing and development environment.
- It improves the availability of servers.
- It allows organizations to make efficient use of resources.
- It reduces redundancy without purchasing additional hardware components.

Storage Virtualization:

Storage virtualization is a process of pooling physical storage devices so that IT may address a single "virtual" storage unit. It offered considerable economic and operational savings over bare metal storage but is now mostly overshadowed by the cloud paradigm.

What is Storage Virtualization?

Storage virtualization is functional RAID levels and controllers are made desirable, which is an important component of storage servers. Applications and operating systems on the device

can directly access the discs for writing. Local storage is configured by the controllers in RAID groups, and the operating system sees the storage based on the configuration. The controller, however, is in charge of figuring out how to write or retrieve the data that the operating system requests because the storage is abstracted.

Types of Storage Virtualization

Below are some types of Storage Virtualization.

- **Kernel-level virtualization:** In hardware virtualization, a different version of the Linux kernel functions. One host may execute several servers thanks to the kernel level.
- **Hypervisor Virtualization: I**nstalled between the operating system and the hardware is a section known as a hypervisor. It enables the effective operation of several operating systems.
- **Hardware-assisted Virtualization:** Hardware-assisted virtualization is similar to complete para-virtualization, however, it needs hardware maintenance.
- **Para-virtualization:** The foundation of para-virtualization is a hypervisor, which handles software emulation and trapping.

Methods of Storage Virtualization

- **Network-based storage virtualization:** The most popular type of virtualization used by businesses is network-based storage virtualization. All of the storage devices in an FC or iSCSI SAN are connected to a network device, such as a smart switch or specially designed server, which displays the network's storage as a single virtual pool.
- **Host-based storage virtualization:** Host-based storage virtualization is software-based and most often seen in HCI systems and cloud storage. In this type of virtualization, the host, or a hyper-converged system made up of multiple hosts, presents virtual drives of varying capacity to the guest machines, whether they are VMs in an enterprise environment, physical servers or computers accessing file shares or cloud storage.
- **Array-based storage virtualization:** Storage using arrays The most popular use of virtualization is when a storage array serves as the main storage controller and is equipped with virtualization software. This allows the array to share storage resources with other arrays and present various physical storage types that can be used as storage tiers.

How Storage Virtualization Works?

- Physical storage hardware is replicated in a virtual volume during storage virtualization.
- A single server is utilized to aggregate several physical discs into a grouping that creates a basic virtual storage system.
- Operating systems and programs can access and use the storage because a virtualization layer separates the physical discs from the virtual volume.
- The physical discs are separated into objects called logical volumes (LV), logical unit numbers (LUNs), or RAID groups, which are collections of tiny data blocks.

- RAID arrays can serve as virtual storage in a more complex setting. Many physical drives simulate a single storage device that copies data to several discs in the background while stripping it.
- The virtualization program has to take an extra step in order to access data from the physical discs.
- Block-level and file-level storage environments can both be used to create virtual storage.

Advantages of Storage Virtualization

- Advanced features like redundancy, replication, and disaster recovery are all possible with the storage devices.
- It enables everyone to establish their own company prospects.
- Data is kept in more practical places that are farther from the particular host. Not always is the data compromised in the event of a host failure.
- IT operations may now provision, divide, and secure storage in a more flexible way by abstracting the storage layer.

Disadvantages of Storage Virtualization

Below are some Disadvantages of Storage Virtualization.

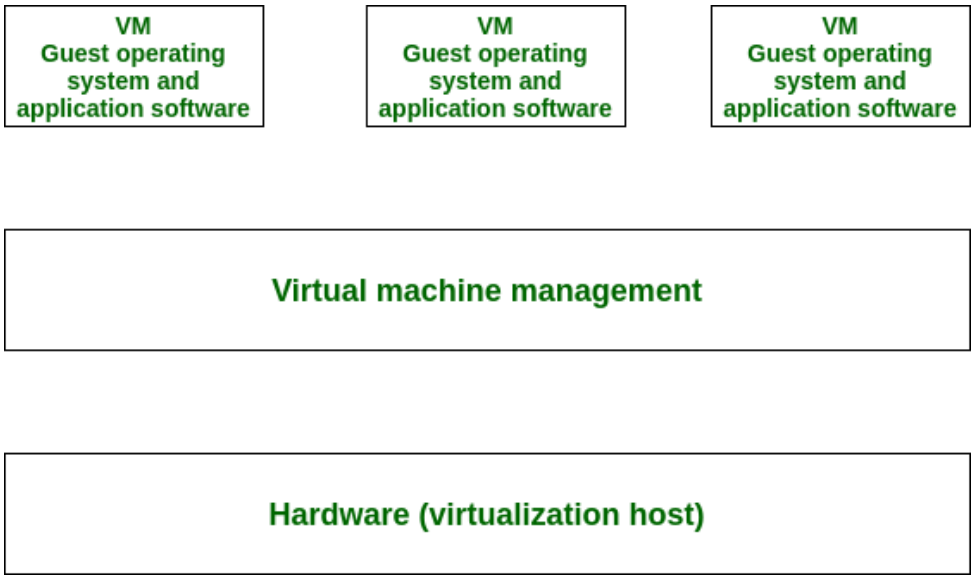
- Storage Virtualization still has limitations which must be considered.
- Data security is still a problem. Virtual environments can draw new types of cyberattacks, despite the fact that some may contend that virtual computers and servers are more secure than physical ones.
- The deployment of storage virtualization is not always easy. There aren't many technological obstacles, including scalability.
- Your data's end-to-end perspective is broken by virtualization. Integrating the virtualized storage solution with current tools and systems is a requirement.

Operating system based Virtualization

Operating system-based Virtualization refers to an operating system feature in which the kernel enables the existence of various isolated user-space instances. The installation of virtualization software also refers to Operating system-based virtualization. It is installed over a pre-existing operating system and that operating system is called the host operating system.

In this virtualization, a user installs the virtualization software in the operating system of his system like any other program and utilizes this application to operate and generate various virtual machines. Here, the virtualization software allows direct access to any of the created virtual machines to the user. As the host OS can provide hardware devices with the mandatory support, operating system virtualization may affect compatibility issues of hardware even when the hardware driver is not allocated to the virtualization software.

Virtualization software is able to convert hardware IT resources that require unique software for operation into virtualized IT resources. As the host OS is a complete operating system in itself, many OS-based services are available as organizational management and administration tools can be utilized for the virtualization host management.



Some major operating system-based services are mentioned below:

1. Backup and Recovery.
2. Security Management.
3. Integration to Directory Services.

Various major operations of Operating System Based Virtualization are described below:

1. Hardware capabilities can be employed, such as the network connection and CPU.
2. Connected peripherals with which it can interact, such as a webcam, printer, keyboard, or Scanners.
3. Data that can be read or written, such as files, folders, and network shares.

The Operating system may have the capability to allow or deny access to such resources based on which the program requests them and the user account in the context of which it runs. OS may also hide these resources, which leads that when a computer program computes them, they do not appear in the enumeration results. Nevertheless, from a programming perspective, the computer program has interacted with those resources and the operating system has managed an act of interaction.

With operating-system-virtualization or containerization, it is probable to run programs within containers, to which only parts of these resources are allocated. A program that is expected to perceive the whole computer, once run inside a container, can only see the allocated resources and believes them to be all that is available. Several containers can be formed on each operating system, to each of which a subset of the computer's resources is

allocated. Each container may include many computer programs. These programs may run parallel or distinctly, even interrelate with each other.

Features of operating system-based virtualization are:

- **Resource isolation:** Operating system-based virtualization provides a high level of resource isolation, which allows each container to have its own set of resources, including CPU, memory, and I/O bandwidth.
- **Lightweight:** Containers are lightweight compared to traditional virtual machines as they share the same host operating system, resulting in faster startup and lower resource usage.
- **Portability:** Containers are highly portable, making it easy to move them from one environment to another without needing to modify the underlying application.
- **Scalability:** Containers can be easily scaled up or down based on the application requirements, allowing applications to be highly responsive to changes in demand.
- **Security:** Containers provide a high level of security by isolating the containerized application from the host operating system and other containers running on the same system.
- **Reduced Overhead:** Containers incur less overhead than traditional virtual machines, as they do not need to emulate a full hardware environment.
- **Easy Management:** Containers are easy to manage, as they can be started, stopped, and monitored using simple commands.

Operating system-based virtualization can raise demands and problems related to performance overhead, such as:

1. The host operating system employs CPU, memory, and other hardware IT resources.
2. Hardware-related calls from guest operating systems need to navigate numerous layers to and from the hardware, which shrinkage overall performance.
3. Licenses are frequently essential for host operating systems, in addition to individual licenses for each of their guest operating systems.

Advantages of Operating System-Based Virtualization:

- **Resource Efficiency:** Operating system-based virtualization allows for greater resource efficiency as containers do not need to emulate a complete hardware environment, which reduces resource overhead.
- **High Scalability:** Containers can be quickly and easily scaled up or down depending on the demand, which makes it easy to respond to changes in the workload.**Easy Management:** Containers are easy to manage as they can be managed through simple commands, which makes it easy to deploy and maintain large numbers of containers. **Reduced Costs:** Operating system-based virtualization can significantly reduce costs, as it requires fewer resources and infrastructure than traditional virtual machines.

- **Faster Deployment:** Containers can be deployed quickly, reducing the time required to launch new applications or update existing ones.
- **Portability:** Containers are highly portable, making it easy to move them from one environment to another without requiring changes to the underlying application.

Disadvantages of Operating System-Based Virtualization:

- **Security:** Operating system-based virtualization may pose security risks as containers share the same host operating system, which means that a security breach in one container could potentially affect all other containers running on the same system.
- **Limited Isolation:** Containers may not provide complete isolation between applications, which can lead to performance degradation or resource contention.
- **Complexity:** Operating system-based virtualization can be complex to set up and manage, requiring specialized skills and knowledge.
- **Dependency Issues:** Containers may have dependency issues with other containers or the host operating system, which can lead to compatibility issues and hinder deployment.
- **Limited Hardware Access:** Containers may have limited access to hardware resources, which can limit their ability to perform certain tasks or applications that require direct hardware access.

Virtualization for Cloud Computing

Cloud computing has transformed the IT landscape, enabling organizations to access scalable, on-demand computing resources over the internet. At the heart of cloud computing lies **virtualization**, a key technology that abstracts physical hardware resources and allows multiple virtual environments to coexist on a single physical system. Virtualization not only optimizes resource utilization but also underpins the flexibility, scalability, and cost-efficiency of modern cloud services.

What is Virtualization?

Virtualization refers to the creation of virtual instances of hardware, software, storage, or networks, allowing multiple workloads to share the same physical resources. This is achieved using a **hypervisor**, a layer of software that sits between the hardware and the virtual machines (VMs), managing their allocation and ensuring isolation.

Role of Virtualization in Cloud Computing

1. **Resource Optimization**: Virtualization enables multiple virtual machines to run on a single server, improving resource utilization by allocating CPU, memory, and storage dynamically based on workload demands.
2. **Scalability**: Virtualization allows rapid scaling of resources by creating or removing virtual machines as needed, making it ideal for elastic workloads in the cloud.
3. **Isolation and Security**: Each virtual machine operates independently, ensuring that issues in one VM do not affect others, providing strong isolation for multi-tenant cloud environments.

4. **Cost Efficiency**: By maximizing resource utilization, virtualization reduces the need for additional physical servers, lowering capital and operational expenses.
5. **Flexibility**: Virtualization supports a wide variety of operating systems and applications, enabling cloud providers to offer diverse services.

Types of Virtualization in Cloud Computing

1. **Server Virtualization**: Divides a physical server into multiple virtual machines, each capable of running its own operating system and applications.
2. **Storage Virtualization**: Aggregates multiple physical storage devices into a single logical storage pool, simplifying data management and access.
3. **Network Virtualization**: Abstracts network resources, creating virtual networks that operate independently of physical infrastructure.
4. **Desktop Virtualization**: Enables users to access their desktops remotely through virtual machines hosted in the cloud.

Benefits of Virtualization in the Cloud

- **Dynamic Resource Allocation**: Virtualization allows efficient management of cloud resources, ensuring workloads receive the resources they need.
- **Disaster Recovery**: Virtual machines can be backed up and migrated easily, ensuring business continuity in case of hardware failures.
- **Energy Efficiency**: By consolidating workloads on fewer servers, virtualization reduces power consumption and carbon footprint.
- **Enhanced Development**: Virtualization enables developers to create isolated test environments, accelerating the development and deployment of cloud-based applications.

Software-Defined Data Center (SDDC)

A software-defined data center (SDDC) is a server management concept in which all infrastructure elements—networking, storage, and compute—are virtualized and delivered as a service. Deployment, operation, provisioning, and configuration are abstracted from hardware, with tasks implemented through software interfaces.

Former VMware CTO Steve Herrod is widely credited with coining the term around 2012. While SDDC is mainly associated with VMware products, the principles of fully virtualized environments have been implemented by various providers.

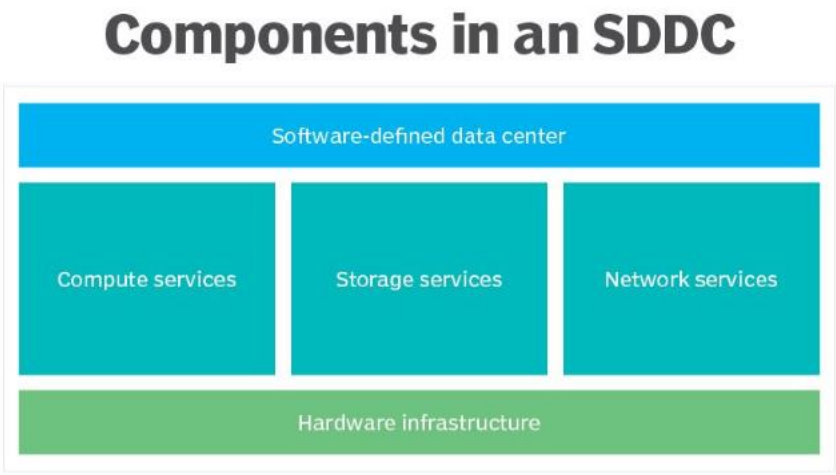
SDDC is increasingly common in enterprise environments and has been adopted by cloud service providers and data-center-as-a-service providers. It is supported by platforms like Amazon, Microsoft Azure, Google, and the Open Compute Project.

In an SDDC, administrators can quickly create and remove entire services with a few simple commands, making it essential for **DevOps** and **infrastructure as code (IaC)** initiatives.

Components of an SDDC

Virtualization is central to the software-defined data center. The three major SDDC building blocks include:

- **Network Virtualization (Software-Defined Networking - SDN)**:
  - Abstracts physical network resources from the logical network topology.
  - Allows combining network resources and splitting available bandwidth into independent channels, which can be assigned or reassigned to servers or devices in real-time.
- **Storage Virtualization**:
  - Pools physical storage from multiple devices into a single logical storage unit managed from a central console.
  - These storage pools can be subdivided into high-performance virtual disks for use by software.
- **Server Virtualization**:
  - Masks server resources (e.g., processors, physical servers, and RAM) from end users, simplifying resource management.
  - Increases resource sharing, utilization, and scalability while maintaining easy expansion.



These components—virtual compute, storage, and network services—are supported by a **business-logic layer** to translate application requests, policies, **and service-level agreements.**

SDDC Benefits and Challenges

- **Benefits**:
  - Dynamically configures and provisions applications, infrastructure, and IT resources as a unified system or aggregated domains.
  - Enables private cloud deployment for better control of services and data.
  - Automates provisioning and management, reducing manual effort and increasing flexibility.
  - Provides agility, elasticity, and scalability similar to public cloud models.
  - Supports resource pooling, enabling workloads to operate independently of physical IT infrastructure.

    - Reduces costs and management overhead while enhancing IT flexibility.
- **Challenges**:
    - Implementation often involves components from multiple vendors, complicating planning and integration.
    - Enterprises must balance flexibility with potential **vendor lock-in** risks.

SDDC Market Share

The concept of SDDC evolved from **x86 server virtualization**, introduced by VMware in 2006. This decoupled processing power and memory from hardware, presenting them as shared resources. Applications now run in virtual machines with full OS copies instead of bare metal.

- VMware vCloud Suite is a key tool for building and managing private clouds based on SDDC architecture.
- The SDDC market includes segments for software-defined compute, networking, and storage.
- According to Fortune Business Insights, the market is projected to grow:
    - **2024**: $71.09 billion (up from $58.04 billion in 2023).
    - **2032**: $307.79 billion, with a compound annual growth rate (CAGR) of 20.1%.

Two approaches exist for transitioning to SDDCs:

1. **Transitional**: Existing hardware runs in parallel with newer SDDC equipment.
2. **Integrated**: Existing and new hardware are unified within a single data center fabric.

Future of SDDC

SDDCs are revolutionizing data center computing by abstracting the application layer from physical hardware, enabling agile development and faster adaptation to business needs.

As SDDCs evolve, enterprises must enhance IT security to address abstraction challenges. Collaboration between security and virtualization teams is crucial to mitigate threats. Software-integrated security allows dynamic policy adaptation based on applications, content, or user behavior.

SDDCs offer a unified virtual infrastructure for seamless resource migration across private, public, and hybrid clouds, ensuring flexibility, scalability, and resilience in modern IT environments.