Cloud-Enabling Technology

Modern-day clouds are underpinned by a set of primary technology components that collectively enable key features and characteristics associated with contemporary cloud computing.

There are lots of technologies that are working behind cloud computing. These type of technologies makes cloud computing more reliable, adaptable, and usable.

The following such technologies are covered in this section:

- Broadband Networks and Internet Architecture
- Data Center Technology
- Virtualization Technology
- Web Technology
- Multitenant Technology
- Containerization

Each existed and matured prior to the advent of cloud computing, although cloud computing advancements helped further evolve select areas of these cloud-enabling technologies.

Broadband Networks and Internet Architecture

- Internet Service Providers (ISPs)
- Connectionless packet switching (datagram networks) and
- Router-based interconnectivity
- Technical and Business Considerations
  - Connectivity issues
  - Network bandwidth and latency issues
  - Cloud carrier and cloud provider selection

Broadband Networks and Internet Architecture

- All clouds must be connected to a network.
- This inevitable requirement forms an inherent dependency on internetworking.
- Internetworks, or the Internet, allow for the remote provisioning of IT resources and are directly supportive of ubiquitous network access.
- Cloud consumers have the option of accessing the cloud using only private and dedicated network links in LANs, although most clouds are Internet-enabled.
- The potential of cloud platforms therefore generally grows in parallel with advancements in Internet connectivity and service quality.
- **Internet Service Providers** (ISPs) Established and deployed by ISPs, the Internet's largest backbone networks are strategically interconnected by core routers that connect the world's multinational networks.
- As shown in Figure 5.1, an ISP network interconnects to other ISP networks and various organizations

- ISPs can freely deploy, operate and manage their networks
- Worldwide connectivity is enabled through a hierarchical topology composed of Tiers 1, 2, and 3 (Figure 5.2).
- The core Tier 1 is made of large-scale, international cloud providers that oversee massive interconnected global networks, which are connected to Tier 2's large regional providers.
- The interconnected ISPs of Tier 2 connect with Tier 1 providers, as well as the local ISPs of Tier 3.
- Cloud consumers and cloud providers can connect directly using a Tier 1 provider, since any operational ISP can enable Internet connection.

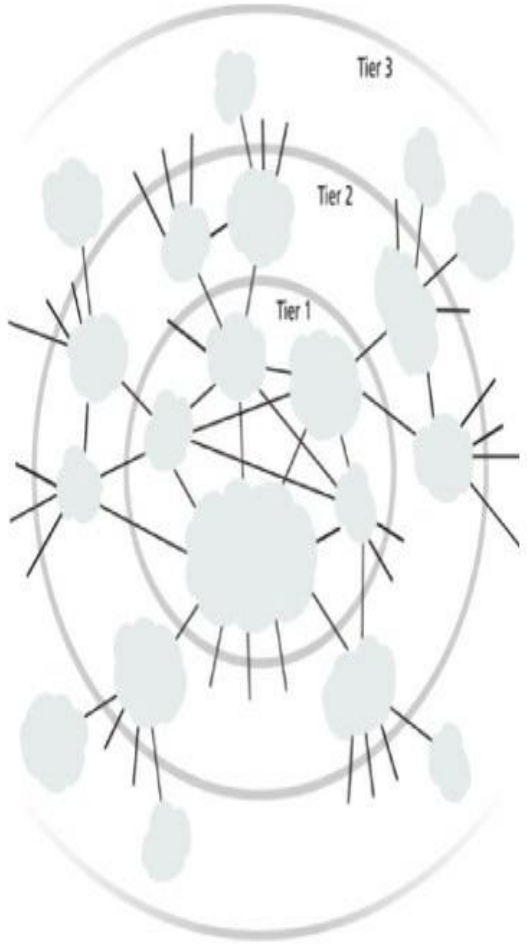


Figure 5.2 An abstraction of the internetworking structure of the Internet.

Two fundamental components used to construct the internetworking architecture are

- connectionless packet switching (datagram networks) and
- router-based interconnectivity
- Connectionless Packet Switching (Datagram Networks) End-to-end (sender-receiver pair) data flows are divided into packets of a limited size that are received and processed through network switches and routers, then queued and forwarded from one intermediary node to the next.
- Each packet carries the necessary location information, such as the Internet Protocol (IP) or Media Access Control (MAC) address, to be processed and routed at every source, intermediary, and destination node.

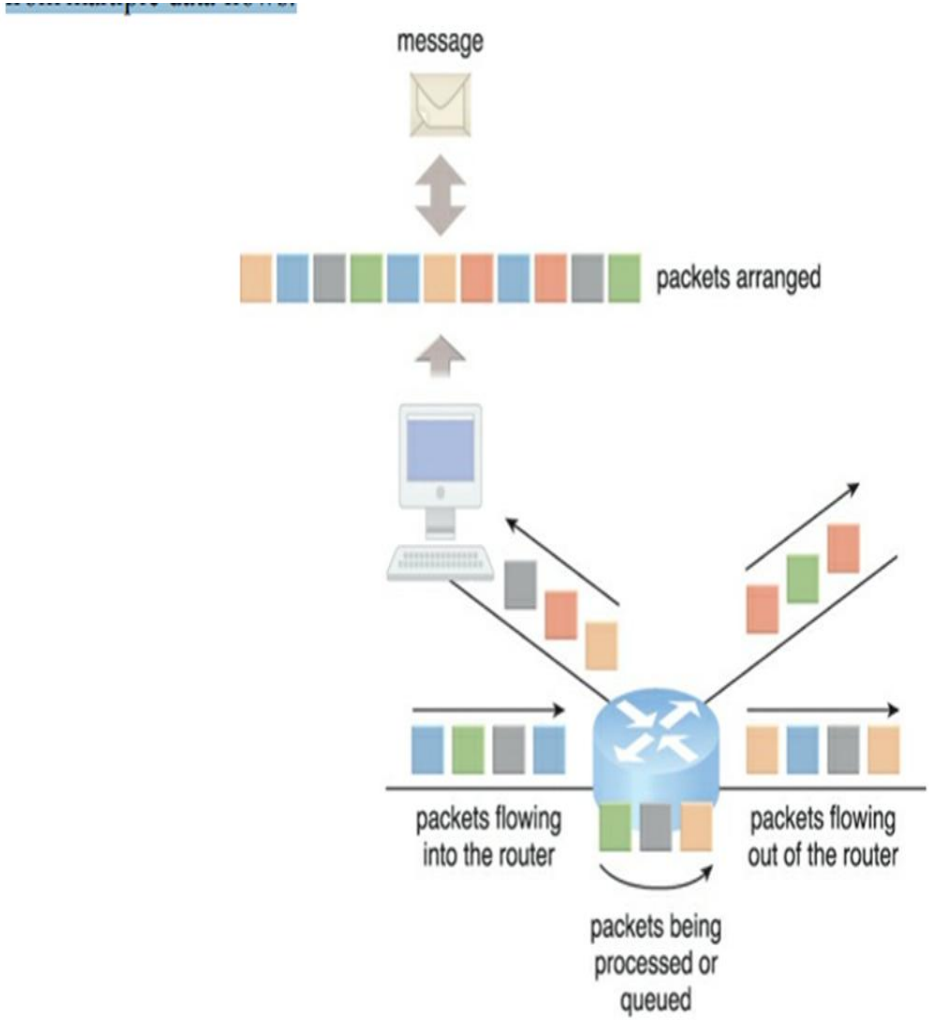


Figure 5.3 Packets traveling through the Internet are directed by a router that arranges them into a message.

Router-Based Interconnectivity

- A router is a device that is connected to multiple networks through which it forwards packets.
- Even when successive packets are part of the same data flow, routers process and forward each packet individually while maintaining the network topology information that locates the next node on the communication path between the source and destination nodes.
- Routers manage **network traffic** and gauge the most efficient hop for packet delivery, since they are privy to both the packet source and packet destination.
- The basic mechanics of internetworking are illustrated in Figure 5.3, in which a message is coalesced from an incoming group of disordered packets.
- The depicted router receives and forwards packets from multiple data flows.

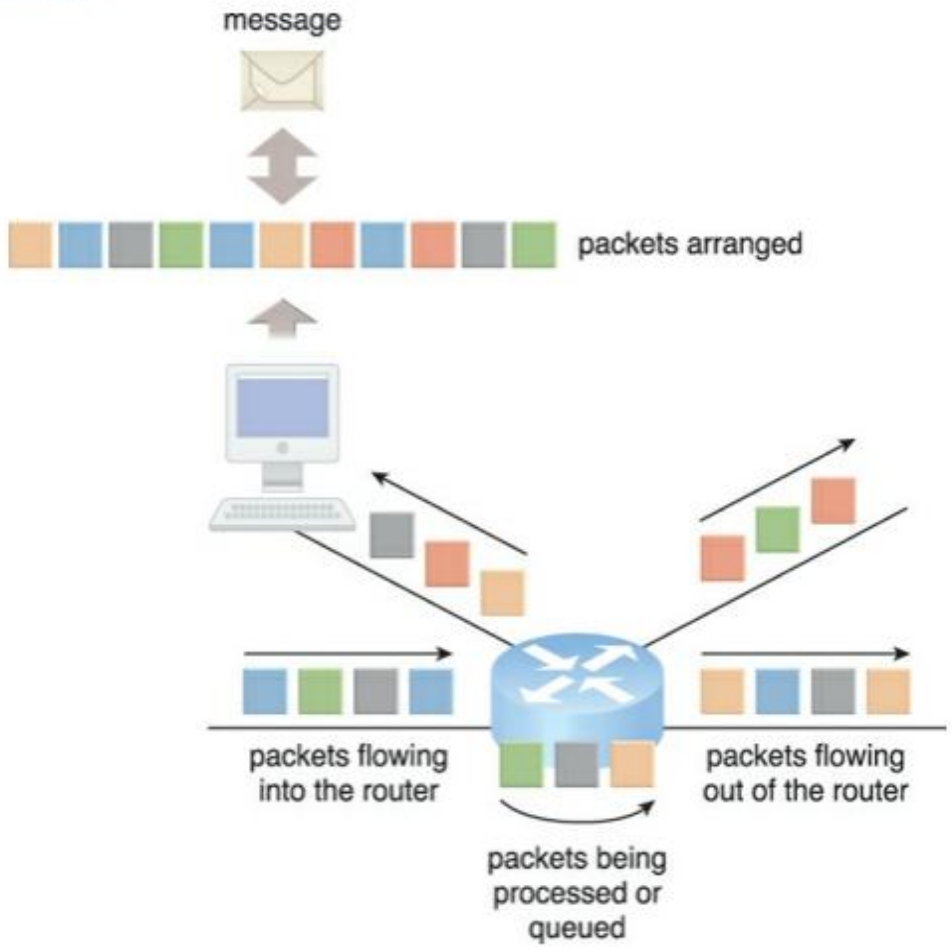


Figure 5.3 Packets traveling through the Internet are directed by a router that arranges them into a message.

- The communication path that connects a cloud consumer with its cloud provider may involve multiple ISP networks.
- The Internet’s mesh structure connects Internet hosts (endpoint systems) using multiple alternative network routes that are determined at runtime.
- Communication can therefore be sustained even during simultaneous network failures, although using multiple network paths can cause routing fluctuations and latency.

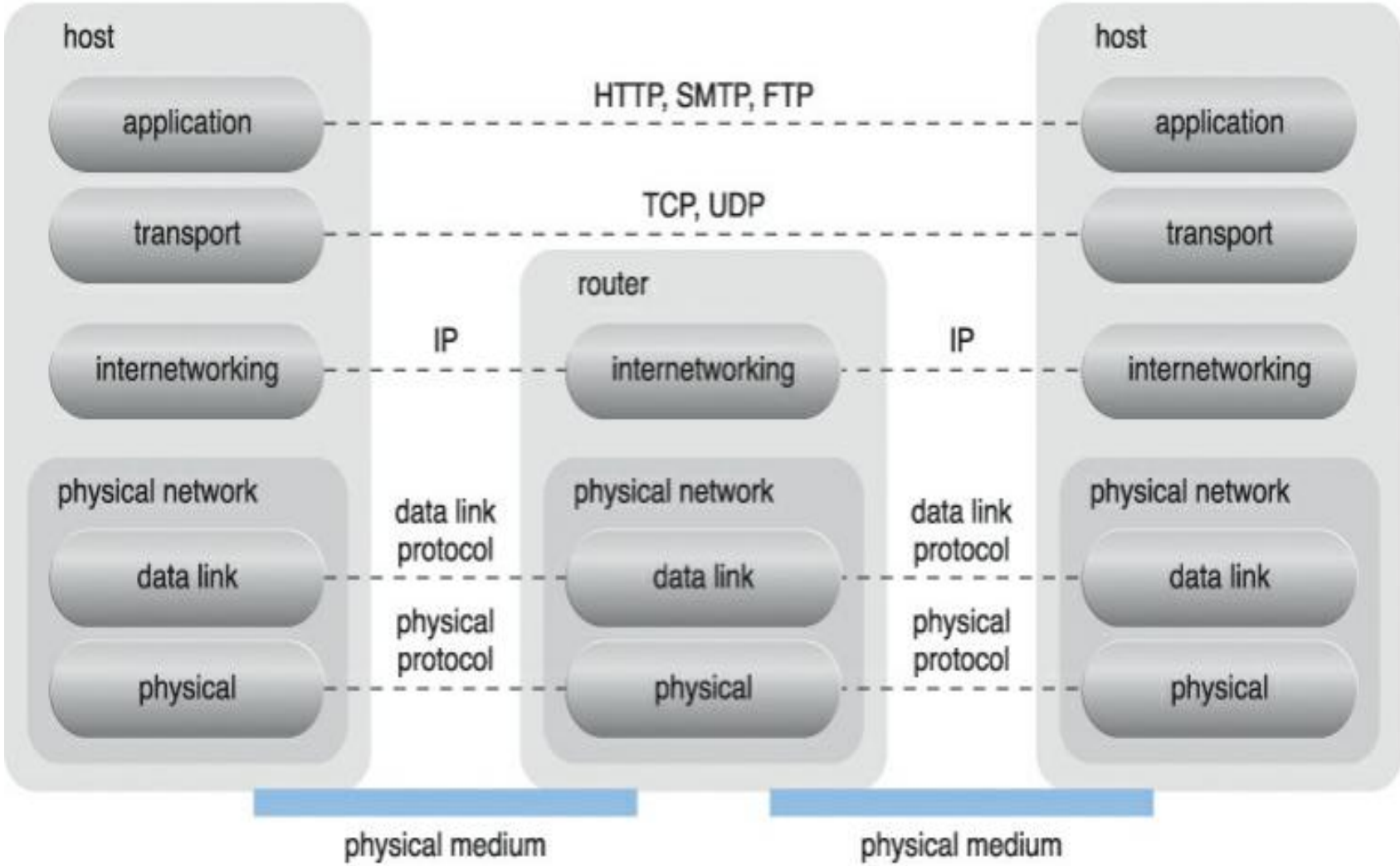


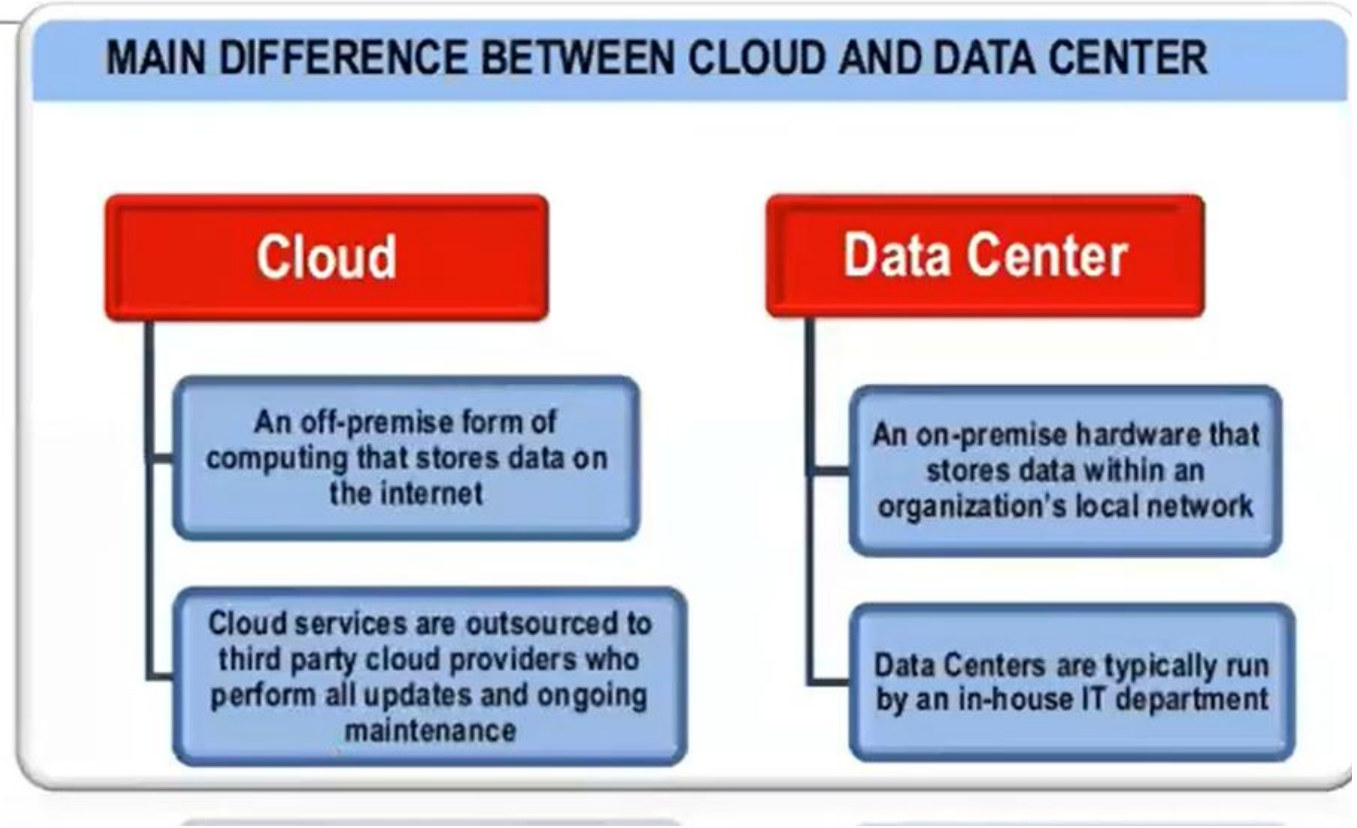
Figure 5.4 A generic view of the Internet reference model and protocol stack.

- This applies to ISPs that implement the Internet's internetworking layer and interact with other network technologies, as follows:
- **Physical Network :**IP packets are transmitted through underlying physical networks that connect adjacent nodes, such as Ethernet, ATM network, and the 3G mobile HSDPA. Physical networks comprise a data link layer that controls data transfer between neighboring nodes, and a physical layer that transmits data bits through both wired and wireless media.
- **Transport Layer Protocol:** Transport layer protocols, such as the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP), use the IP to provide standardized, end-to-end communication support that facilitates the navigation of data packets across the Internet.
- **Application Layer Protocol:** Protocols such as HTTP, SMTP for e-mail, BitTorrent for P2P, and SIP for IP telephony use transport layer protocols to standardize and enable specific data packet transferring methods over the Internet. Many other protocols also fulfill application-centric requirements and use either TCP/IP or UDP as their primary method of data transferring across the Internet and LANs

Data Center Technology

- Grouping IT resources in close proximity with one another, rather than having them geographically dispersed, allows for power sharing, higher efficiency in shared IT resource usage, and improved accessibility for IT personnel.
- These are the advantages that naturally popularized the data center concept.
- Modern data centers exist as specialized IT infrastructure used to house centralized IT resources, such as servers, databases, networking and telecommunication devices, and software systems.

- *Data centers* exist as specialized IT infrastructure used to house centralized IT resources, such as Servers, Databases, Networking and Telecommunication devices, and Software systems.



Data centers are typically comprised of the following technologies and components:

- Virtualization
- Standardization and Modularity
- Automation
- Remote Operation and Management
- High Availability
- Security-Aware Design, Operation, and Management
- Facilities
- Computing Hardware
- Storage Hardware
- Network Hardware

Virtualization

- Data centers consist of both physical and virtualized IT resources.
- The physical IT resource layer refers to the facility infrastructure that houses computing/networking systems and equipment, together with hardware systems and their operating systems (Figure 5.7).
- The resource abstraction and control of the virtualization layer is comprised of operational and management tools that are often based on virtualization platforms that abstract the physical computing and networking IT resources as virtualized components that are easier to allocate, operate, release, monitor, and control

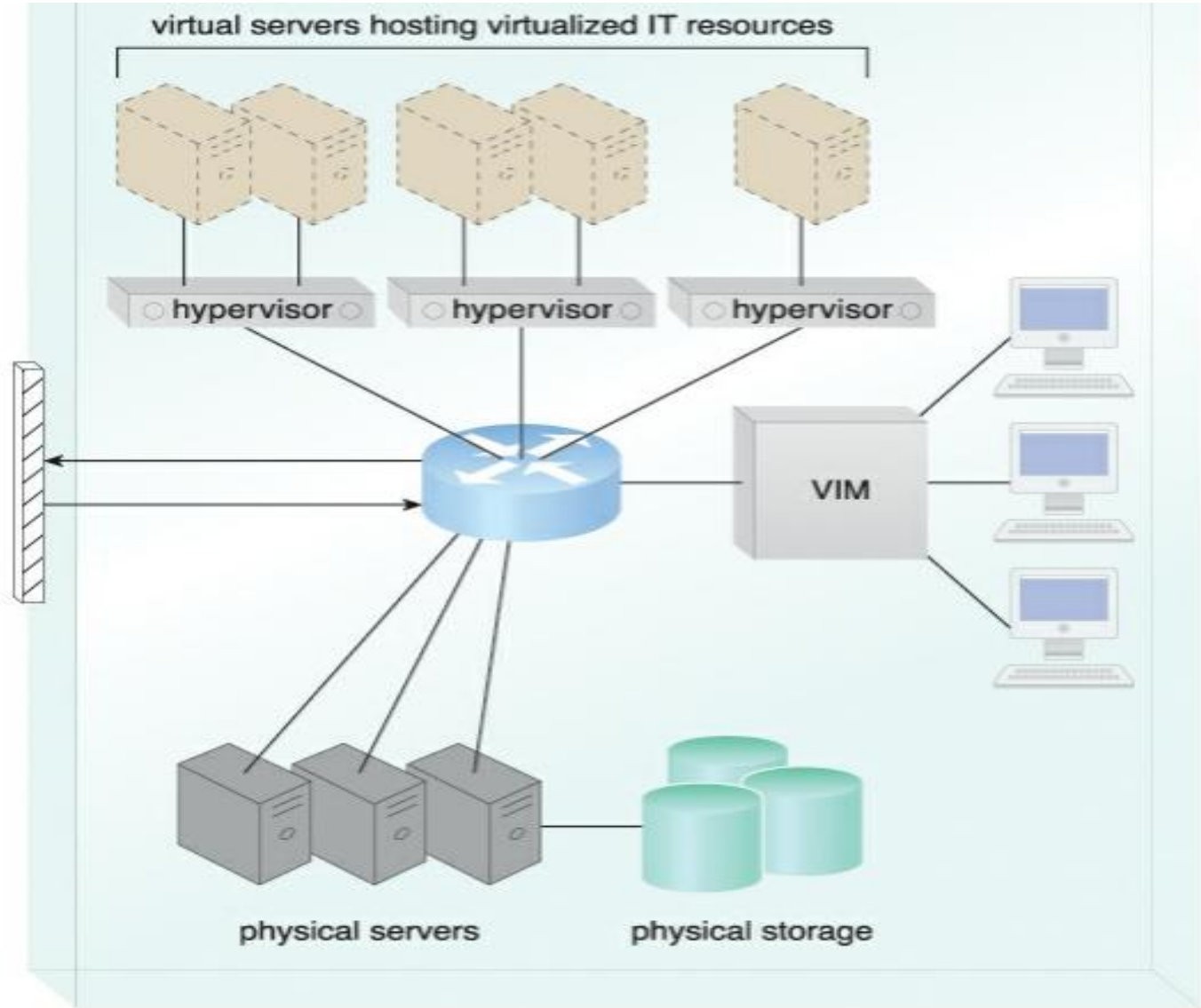
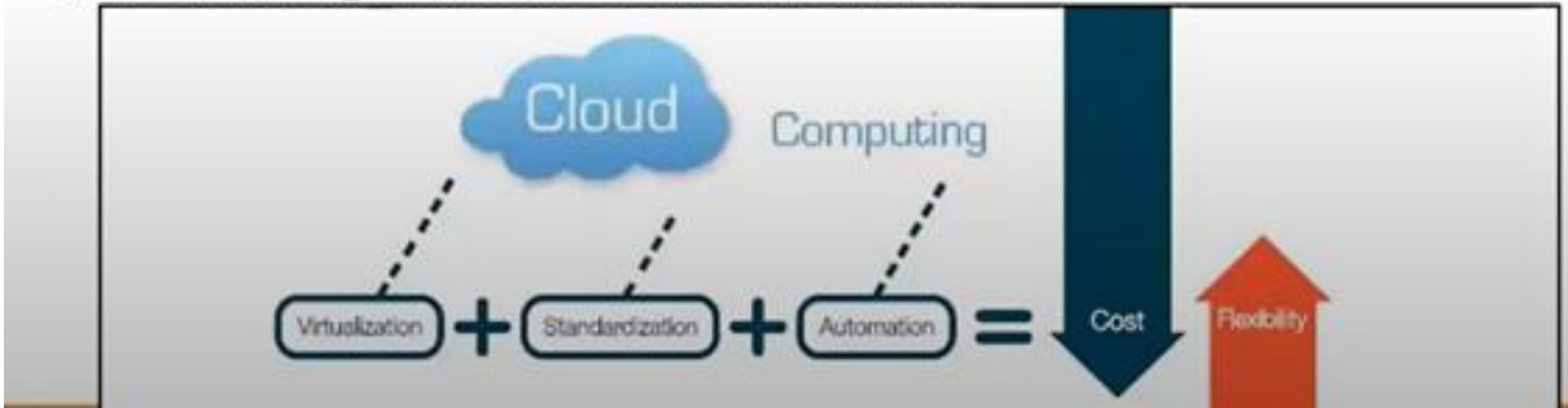


Figure 5.7 The common components of a data center working together to provide virtualized IT resources supported by physical IT resources.

Standardization and Modularity

- Data centers are built upon standardized commodity hardware and designed with modular architectures, aggregating multiple identical building blocks of facility infrastructure and equipment to support scalability, growth, and speedy hardware replacements.
- Modularity and standardization are key requirements for reducing investment and operational costs as they enable economies of scale for the procurement, acquisition, deployment, operation, and maintenance processes.
- Common virtualization strategies and the constantly improving capacity and performance of physical devices both favor IT resource consolidation, since fewer physical components are needed to support complex configurations. Consolidated IT resources can serve different systems and be shared among different cloud consumer

- Modularity and standardization are key requirements for reducing investment and operational costs as they enable economies of scale for the **procurement**, **acquisition**, **deployment**, **operation**, and **maintenance** processes.
- Consolidated IT resources can serve different systems and be shared among different cloud consumers.



Automation

- Data centers have specialized platforms that automate tasks like provisioning, configuration, patching, and monitoring without supervision.
- Advances in data center management platforms and tools leverage autonomic computing technologies to enable self-configuration and self-recovery.

Remote Operation and Management

- Most of the operational and administrative tasks of IT resources in data centers are commanded through the network's remote consoles and management systems.
- Technical personnel are not required to visit the dedicated rooms that house servers, except to perform highly specific tasks, such as equipment handling and cabling or hardware-level installation and maintenance.

High Availability

- Since any form of data center outage significantly impacts business continuity for the organizations that use their services, data centers are designed to operate with increasingly higher levels of redundancy to sustain availability.
- Data centers usually have redundant, uninterruptable power supplies, cabling, and environmental control subsystems in anticipation of system failure, along with communication links and clustered hardware for load balancing.

Security-Aware Design, Operation, and Management

- Requirements for security, such as physical and logical access controls and data recovery strategies, need to be thorough and comprehensive for data centers, since they are centralized structures that store and process business data.
- Due to the sometimes prohibitive nature of building and operating on-premise data centers, outsourcing data center-based IT resources has been a common industry practice for decades.
- However, the outsourcing models often required long-term consumer commitment and usually could not provide elasticity, issues that a typical cloud can address via inherent features, such as ubiquitous access, on-demand provisioning, rapid elasticity, and pay-per-use.

Facilities

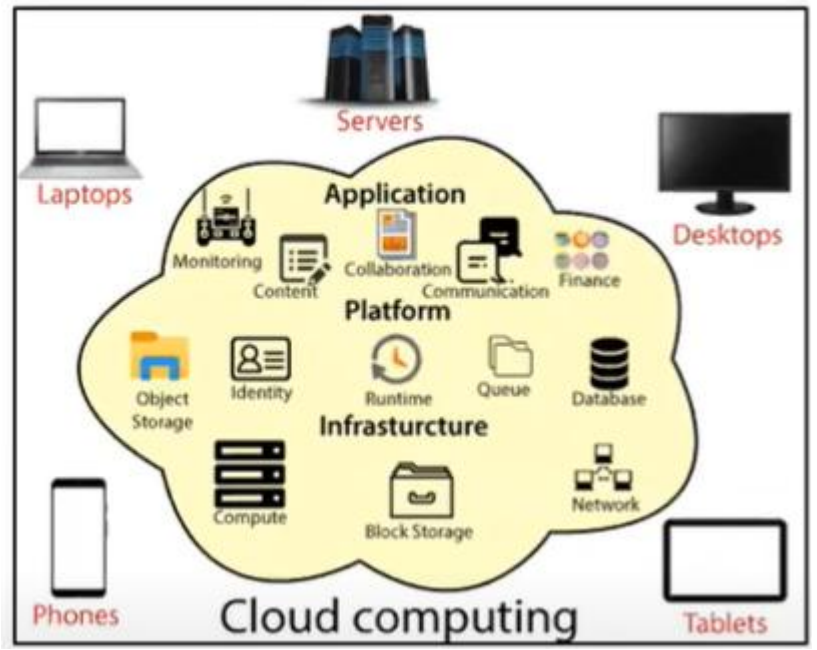
- Data center facilities are custom-designed locations that are outfitted with specialized computing, storage, and network equipment.
- These facilities have several functional layout areas, as well as various power supplies, cabling, and environmental control stations that regulate heating, ventilation, air conditioning, fire protection, and other related subsystems.

Computing Hardware

Much of the heavy processing in data centers is often executed by standardized commodity servers that have substantial computing power and storage capacity.

Several computing hardware technologies are integrated into these modular servers, such as:

- rackmount form factor server design composed of standardized racks with interconnects for power, network, and internal cooling
- support for different hardware processing architectures, such as x86-32bits, x86-64, and RISC
- a power-efficient multi-core CPU architecture that houses hundreds of processing cores in a space as small as a single unit of standardized racks
- redundant and hot-swappable components, such as hard disks, power supplies, network interfaces, and storage controller cards



- Computing architectures such as blade server technologies use rack-embedded physical interconnections (blade enclosures), fabrics (switches), and shared power supply units and cooling fans.
- The interconnections enhance inter-component networking and management while optimizing physical space and power.
- These systems typically support individual server hot-swapping, scaling, replacement, and maintenance, which benefits the deployment of fault-tolerant systems that are based on computer clusters.
- Contemporary computing hardware platforms generally support industry-standard and proprietary operational and management software systems that configure, monitor, and control hardware IT resources from remote management consoles. With a properly established management console, a operator can oversee hundreds to thousands of physical servers, virtual servers, and other IT resources.

Storage Hardware

Data centers have specialized storage systems that maintain enormous amounts of digital information in order to fulfill considerable storage capacity needs.

These storage systems are containers housing numerous hard disks that are organized into arrays. Storage systems usually involve the following technologies:

- Hard Disk Arrays – These arrays inherently divide and replicate data among multiple physical drives, and increase performance and redundancy by including spare disks. This technology is often implemented using redundant arrays of independent disks (RAID) schemes, which are typically realized through hardware disk array controllers.
- I/O Caching – This is generally performed through hard disk array controllers, which enhance disk access times and performance by data caching.
- Hot-Swappable Hard Disks – These can be safely removed from arrays without requiring prior powering down.
- Storage Virtualization – This is realized through the use of virtualized hard disks and storage sharing.
- Fast Data Replication Mechanisms – These include snapshotting, which is saving a virtual machine's memory into a hypervisor-readable file for future reloading, and volume cloning, which is copying virtual or physical hard disk volumes and partitions. Storage systems encompass tertiary redundancies, such as robotized tape libraries, which are used as backup and recovery systems that typically rely on removable media. This type of system can exist as a networked IT resource or direct-attached storage (DAS), in which a storage system is directly connected to the computing IT resource using a host bus adapter (HBA). In the former case, the storage system is connected to one or more IT resources through a network.

Networked storage devices usually fall into one of the following categories:

- Storage Area Network (SAN) – Physical data storage media are connected through a dedicated network and provide block-level data storage access using industry standard protocols, such as the Small Computer System Interface (SCSI).
- Network-Attached Storage (NAS) – Hard drive arrays are contained and managed by this dedicated device, which connects through a network and facilitates access to data using filecentric data access protocols like the Network File System (NFS) or Server Message Block (SMB). NAS, SAN, and other more advanced storage system options provide fault tolerance in many components through controller redundancy, cooling redundancy, and hard disk arrays that use RAID storage technology.

Network Hardware

- Data centers require extensive network hardware in order to enable multiple levels of connectivity. For a simplified version of networking infrastructure, the data center is broken down into five network subsystems, followed by a summary of the most common elements used for their implementation.

Carrier and External Networks Interconnection

- A subsystem related to the internetworking infrastructure, this interconnection is usually comprised of backbone routers that provide routing between external WAN connections and the data center's LAN, as well as perimeter network security devices such as firewalls and VPN gateways.
- Web-Tier Load Balancing and Acceleration This subsystem comprises Web acceleration devices, such as XMLpre-processors, encryption/decryption appliances, and layer 7 switching devices that perform content-aware routing.
- LAN Fabric The LAN fabric constitutes the internal LAN and provides high-performance and redundant connectivity for all of the data center's network-enabled IT resources. It is often implemented with multiple network switches that facilitate network communications and operate at speeds of up to ten gigabits per second. These advanced network switches can also perform several virtualization functions, such as LAN segregation into VLANs, link aggregation, controlled routing between networks, load balancing, and failover.

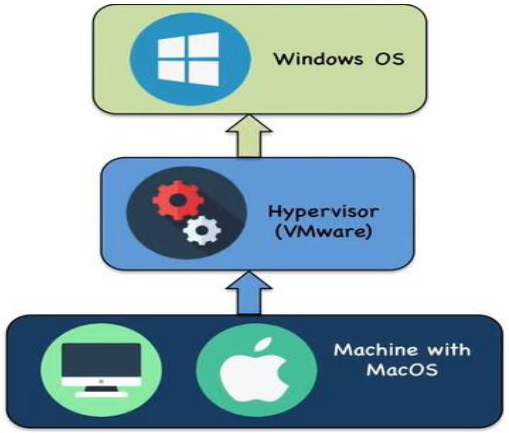
- **SAN Fabric** Related to the implementation of storage area networks (SANs) that provide connectivity between servers and storage systems, the SAN fabric is usually implemented with Fibre Channel (FC), Fibre Channel over Ethernet (FCoE), and InfiniBand network switches.
- **NAS Gateways** This subsystem supplies attachment points for NAS-based storage devices and implements protocol conversion hardware that facilitates data transmission between SAN and NAS devices.
- Data center network technologies have operational requirements for scalability and high availability that are fulfilled by employing redundant and/or fault-tolerant configurations. These five network subsystems improve data center redundancy and reliability to ensure that they have enough IT resources to maintain a certain level of service even in the face of multiple failures.
- Ultra high-speed network optical links can be used to aggregate individual gigabit-per-second channels into single optical fibers using multiplexing technologies like dense wavelength-division multiplexing (DWDM).
- Spread over multiple locations and used to interconnect server farms, storage systems, and replicated data centers, optical links improve transfer speeds and resiliency

Virtualization Technology

Virtualization is the process of converting a physical IT resource into a virtual IT resource. Most types of IT resources can be virtualized, including:

- Servers – A physical server can be abstracted into a virtual server.
- Storage – A physical storage device can be abstracted into a virtual storage device or a virtual disk.
- Network – Physical routers and switches can be abstracted into logical network fabrics, such as VLANs.
- Power – A physical UPS and power distribution units can be abstracted into what are commonly referred to as virtual UPSs.

This section focuses on the creation and deployment of virtual servers through server virtualization technology



- Host or physical host

  Virtualization software runs on a physical server called a host or physical host, whose underlying hardware is made accessible by the virtualization software

- Hypervisor or virtual machine monitor (VMM)
- The virtualization software functionality encompasses system services that are specifically related to virtual machine management and not normally found on standard operating systems

Hardware Independence

- The installation of an operating system's configuration and application software in a unique IT hardware platform results in many software-hardware dependencies.
- In a non-virtualized environment, the operating system is configured for specific hardware models and requires reconfiguration if these IT resources need to be modified.
- Virtualization is a conversion process that translates unique IT hardware into emulated and standardized software-based copies.
- Through hardware independence, virtual servers can easily be moved to another virtualization host, automatically resolving multiple hardware-software incompatibility issues.
- As a result, cloning and manipulating virtual IT resources is much easier than duplicating physical hardware.

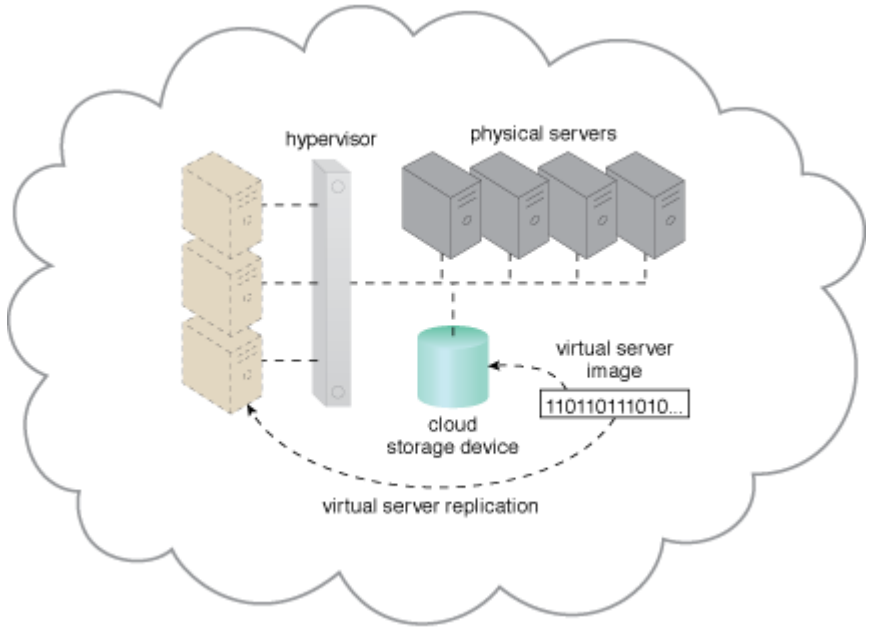
server consolidation

- The coordination function that is provided by the virtualization software allows multiple virtual servers to be simultaneously created in the same virtualization host.
- Virtualization technology enables different virtual servers to share one physical server. This process is called server consolidation and is commonly used to
  - increase hardware utilization,
  - load balancing,
  - and optimization of available IT resources.

The resulting flexibility is such that different virtual servers can run different guest operating systems on the same host.

Resource replication

- Resource replication is defined as the creation of multiple instances of the same IT resource
- Virtualization technology is used to implement the resource replication mechanism to replicate cloud-based IT resources
- The resource replication mechanism is commonly implemented as a hypervisor. For example, the virtualization platform's hypervisor can access a virtual server image to create several instances, or to deploy and replicate ready-made environments and entire applications.



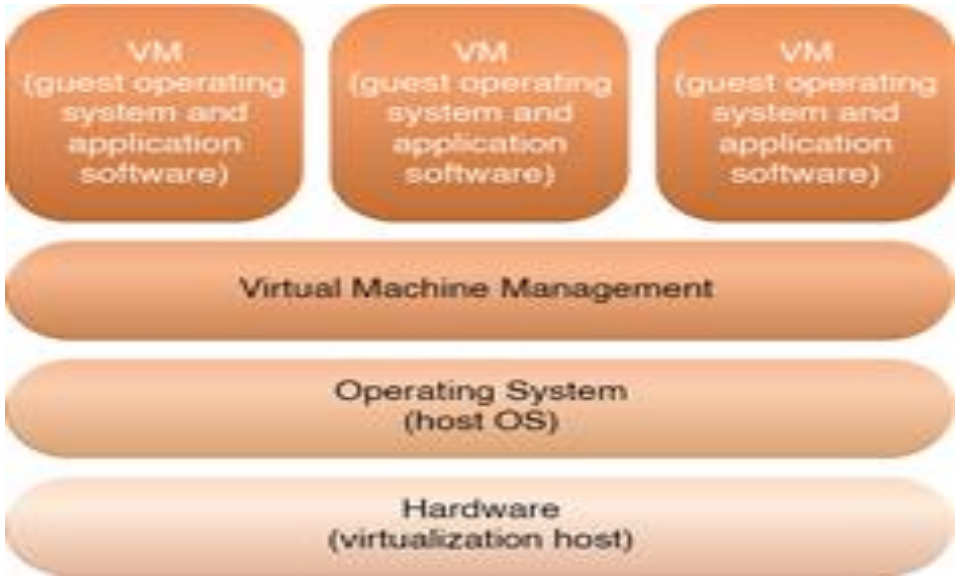
Virtualization types

- Operating System-Based Virtualization
- Hardware-Based Virtualization

Operating System-based Virtualization

- Operating system-based virtualization is the installation of virtualization software in a pre-existing operating system, which is called the *host operating system* (Figure 1).
- For example, a user whose workstation has a specific version of Windows installed decides it wants to generate virtual machines.
- It installs the virtualization software into its host operating system like any other program and uses this application to generate and operate one or more virtual machine.
- This user needs to use its virtualization software to enable direct access to any of the generated virtual machines. Since the host operating system can provide hardware devices with the necessary support, operating system virtualization can rectify hardware compatibility issues even if the hardware driver is unavailable to the virtualization software.

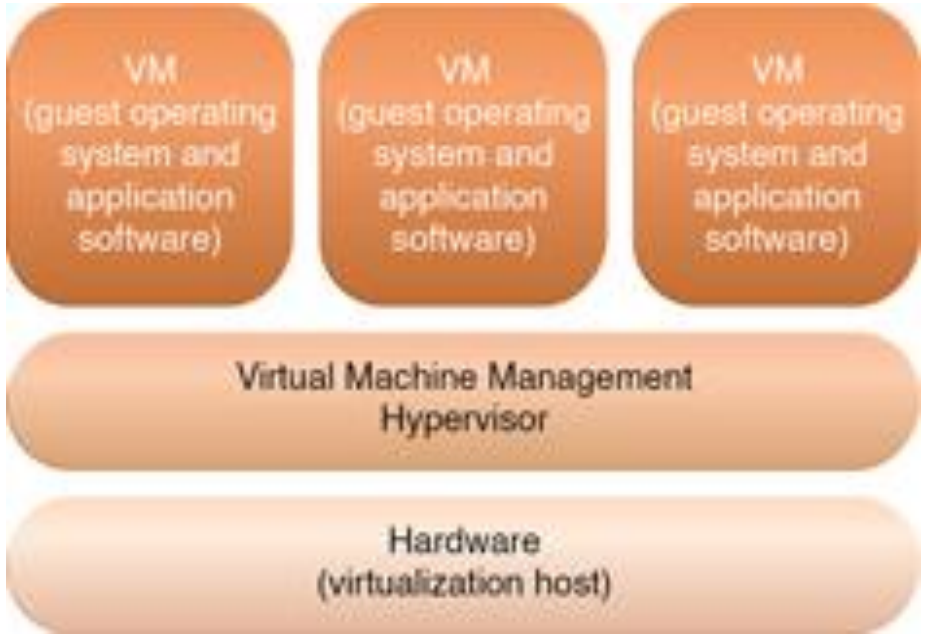
- Hardware independence that is enabled by virtualization allows hardware IT resources to be more flexibly used.
- For example, let’s take a scenario in which the host operating system has the software necessary for controlling five network adapters that are available to the physical computer.
- The virtualization software can make the five network adapters available to the virtual machine, even if the virtualized operating system is usually incapable of physically housing five network adapters.



- Virtualization software translates hardware IT resources that require unique software for operation into virtualized IT resources that are compatible with a range of operating systems. Since the host operating system is a complete operating system in itself, many operating system-based services that are available as organizational management and administration tools can be used to manage the virtualization host.
- Examples of such services include:
- Backup and Recovery
- Integration to Directory Services
- Security Management

Hardware-based Virtualization

- This option represents the installation of virtualization software directly on the virtualization host hardware so as to bypass the host operating system, which would presumably be engaged with operating system-based virtualization (Figure 2).
- Allowing the virtual machines to interact with hardware without requiring intermediary action from the host operating system generally makes hardware-based virtualization more efficient.



The different logical layers of hardware-based virtualization, which does not require another host operating system.

Web Technology

Due to cloud computing's fundamental reliance on internetworking,

- Web browser universality,
- and the ease of Web-based service development,

Web technology is generally used as both the

- implementation medium
- and the management interface for cloud services.

This section introduces the primary Web technologies and discusses their relationship to cloud services

- Basic Web Technology
- Web Applications

Basic Web Technology

The World Wide Web is a **system of interlinked IT resources that are accessed through the Internet**.

The two basic components of the Web are the Web browser

- client
- and the Web server.

Other components, such as

- proxies,
- caching services,
- gateways,
- and load balancers,

are used to improve Web application characteristics such as **scalability and security**.

These additional components reside in a layered architecture that is positioned between the client and the server

Three fundamental elements comprise the technology architecture of the Web:

- **Uniform Resource Locator (URL)** – A standard syntax used for creating identifiers that point to Web-based resources, the URLis often structured using a logical network location.
- **Hypertext Transfer Protocol (HTTP)** – This is the primary communications protocol used to exchange content and data throughout the World Wide Web. URLs are typically transmitted via HTTP.
- **Markup Languages (HTML, XML)** – Markup languages provide a lightweight means of expressing Web-centric data and metadata. The two primary markup languages are HTML (which is used to express the presentation of Web pages) and XML(which allows for the definition of vocabularies used to associate meaning to Web-based data via metadata).

For example, a Web browser can request to execute an action like

- read
- write
- update
- or delete on a Web resource on the Internet
- and proceed to identify and locate the Web resource through its URL.

The request is sent using HTTP to the resource host, which is also identified by a URL.

The Web server locates the Web resource and performs the requested operation, which is followed by a response being sent back to the client.

The response may be comprised of content that includes HTML and XML statements.

Web resources are represented as hypermedia as opposed to hypertext, meaning media such as

- graphics,
- audio,
- video,
- plain text,
- and URLs can be referenced collectively in a single document.

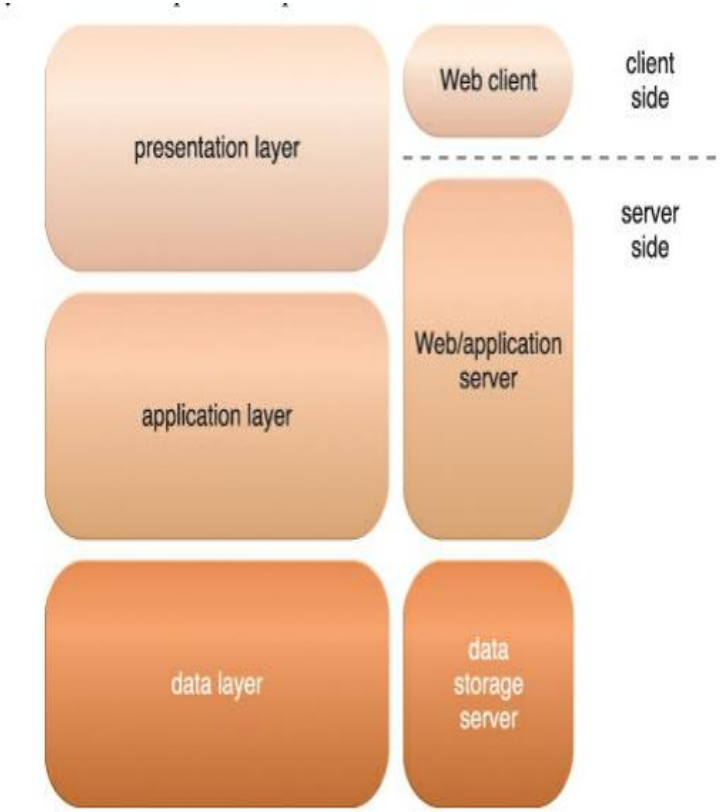
Web Applications

A distributed application that uses Web-based technologies (and generally relies on Web browsers for the presentation of user-interfaces) is typically considered a Web application.

These applications can be found in all kinds of cloud-based environments due to their high accessibility.

presents a common architectural abstraction for Web applications that is based on the basic three-tier model.

- The first tier is called the presentation layer, which represents the user interface.
- The middle tier is the application layer that implements application logic
- the third tier is the data layer that is comprised of persistent data stores.



- The presentation layer has components on both the client and server-side.
- Web servers receive client requests and retrieve requested resources directly as static Web content and indirectly as dynamic Web content, which is generated according to the application logic.
- Web servers interact with application servers in order to execute the requested application logic, which then typically involves interaction with one or more underlying databases.
- PaaS ready-made environments enable cloud consumers to develop and deploy Web applications.
- Typical PaaS offerings have separate instances of the Web server, application server, and data storage server environments.

Containerization

Containerization is an operating system-level virtualization technology used to deploy and run applications and cloud services without the need to deploy a virtual server for each solution.

- Instead, they are deployed within containers.
- Using containers enables multiple isolated cloud services to run on a single physical server or virtual server while accessing the same operating system kernel.
- The operating system kernel allows for the existence of multiple isolated user-space instances or multiple isolated runtimes known as containers, partitions, virtual engines, jails or chroot jails.

Containers are packages of software that contain all of the necessary elements to run in any environment

Containers are lightweight packages of your application code together with dependencies such as specific versions of programming language runtimes and libraries required to run your software services.

Containers make it easy to share CPU, memory, storage, and network resources at the operating systems level and offer a logical packaging mechanism in which applications can be abstracted from the environment in which they actually run.

From Gmail to YouTube to Search, everything at Google runs in containers.

- Regardless of which runtime is used, when a cloud service executes within a container, it is running on a real computer from its point of view.
- A cloud service running on a physical or virtual server operating system can see all of the provided resources, such as connected devices, ports, files, folders, network shares, CPUs, as well as the physical addressable memory.
- However, a cloud service running inside a container can only see the container’s contents and devices attached to the container

Containerization Vs. Virtualization

- As explained earlier, virtualization refers to the act of creating a virtual, rather than an actual version of something.
- This includes virtual computer hardware platforms, storage devices, and computer network resources.
- Virtual servers are an abstraction of physical hardware via server visualization and the use of hypervisors for abstracting a given physical server into multiple virtual servers.

- The hypervisor allows multiple virtual servers to run on a single physical host.
- Virtual servers see the emulated hardware presented to them by the hypervisor as real hardware, and each virtual server has its own operating system, also known as a guest operating system, that needs to be deployed inside the virtual server and managed and maintained as if it were deployed on a physical server.

- In contrast, containers are an abstraction at the application or service layer that package code and dependencies together.
- Multiple containers can be deployed on the same machine and share an operating system kernel with other containers.
- Each container runs as an isolated process in the user space. Containers do not require the guest operating system that is needed for virtual servers and can run directly on a physical server’s operating system.
- Containers also consume less storage space than virtual servers. Figure 5.12 depicts the difference between virtual servers and containers.

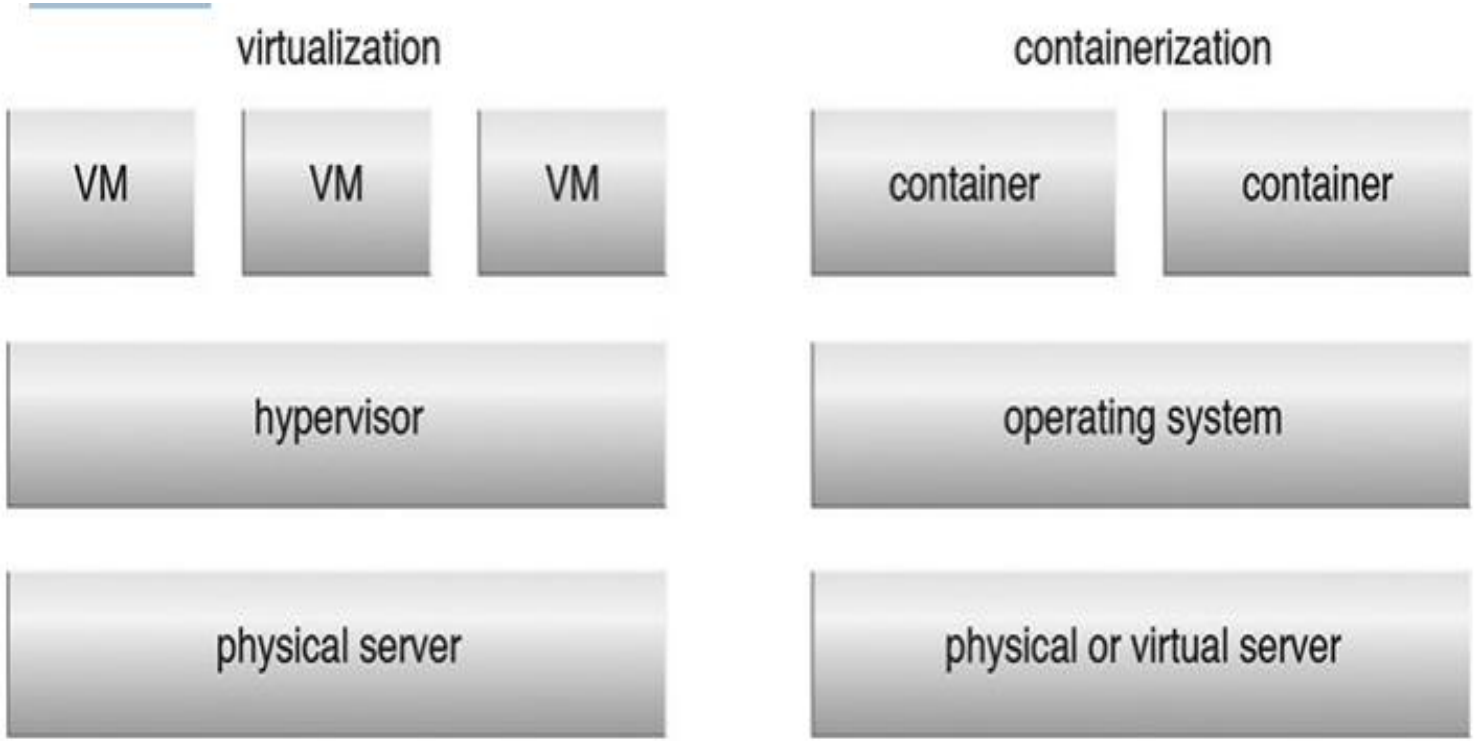


Figure 5.12. A comparison of server virtualization and containerization.

Containers can be deployed in virtual servers, in which case nested virtualization is required to allow the container engine to be installed and operated. Nested virtualization refers to the deployment where one virtualized system is deployed on another.

Benefits of Containers

- Portability is one of the key benefits of containers, allowing cloud resource administrators to move containers to any environment that shares the same host operating system and container engine that the container is hosted on, and without the need to change the application or software (which would normally require changes to the source code).
- Efficient resource utilization is achieved by significantly reducing the CPU, memory and storage usage footprint compared to virtual servers. It is possible to support several containers on the same infrastructure required by a single virtual server, resulting in performance improvements.
- Containers can be created and deployed much faster than virtual servers, which supports a more agile process and facilitates continuous integration.
- Furthermore, containers allow versions of an software code and its dependencies to be tracked.
- Some container images provide the capability of a manifest file that allows cloud service owners and developers to maintain and track versions of a container and its software, inspect differences between different the versions, and roll back to previous versions, when required.

Fundamental Container Architecture Elements

- Container Engine
- Container Build File
- Container Image
- Container
- Networking Address
- Storage Device

Container Engine

- The key component of container architecture is the container engine, also referred to as the containerization engine.
- The container engine is specialized software that is deployed in an operating system to abstract the required resources and enable the definition and deployment of containers.
- Container engine software can be deployed on physical machines or virtual machines. Each container engine provides a set of management tools and commands/APIs to create, modify, schedule, run, stop, start or delete the containers

Container Build File

- A container build file is a descriptor (created by the user or service) that represents the requirements of the application and services that run inside the container, as well as the configuration parameters required by the container engine in order to create and deploy the container.
- The syntax and format of the container build file and configuration parameters it defines depend on the choice of container engine.

Container Image

- The container engine uses a container image to deploy an image based on pre-defined requirements.
- For example, if an application requires a database component or Web server service to operate, these requirements are defined by the user in the container build file. Based on the defined descriptions, the container engine customizes the operating system image and the required commands or services for the application.
- This customized image is normally an immutable read-only image, which enables the deployed application or services in the container to function and perform tasks, but prevents any changes from being made.

Container

- The container is an executable instance of a pre-defined or customized container image that contains one or more software programs, most commonly an application or service.
- While containers are isolated from each other, they may be required to access a shared resource over the network, such as a file system or remote IT resource. This is possible without impacting the isolated containers.
- Each container may have one application or process running in it. Containers can also host multiple applications, services or processes.
- Applications deployed in a container are typically scheduled with the container, meaning that they start and stop with the container.

Networking Address

- Each container has its own network address (such as an IP address) used to communicate with other containers and external components.
- A container can be connected to more than one network by allocating additional network addresses to the container.
- Containers use the physical or virtual network card of the system that the container engine is deployed on to communicate with other containers and IT resources.
- When multiple applications need to be deployed and isolated, containers are used to isolate the applications from each other while still sharing an IP address, the containers can be deployed in a pod.
- Though the sharing of storage devices between containers within a pod is optional, all containers inside the pod share the same IP address.

Storage Device

- Similar to the networking address, a container may connect to one or more storage devices that are made available to the containers over the network.
- Each container has its own level of access to the storages defined by the system or administrators.