Information Storage and Management Version 3

Support Contact: Education Services

Course Overview

Description	Information Storage and Management (ISM) is a unique course that provides a comprehensive understanding of the various storage infrastructure components in data center environments. It enables participants to make informed decisions on storage-related technologies in an increasingly complex IT environment, which is fast changing with the adoption of software-defined infrastructure management and third platform technologies (cloud, Big Data, social, and mobile technologies). It provides a strong understanding of storage technologies and prepares participants for advanced concepts, technologies, and processes. Participants will learn the architectures, features, and benefits of intelligent storage systems including block-based, file-based, object-based, and unified storage; software-defined storage; storage networking technologies such as FC SAN, IP SAN, and FCoE SAN; business continuity solutions such as backup and replication; the highly-critical areas of information security; and storage infrastructure management. This course takes an open-approach to describe all the concepts and technologies, which are further illustrated and reinforced with EMC-related product examples.
Objectives	Upon completion of this course, you should be able to: • Describe data center infrastructure and its elements • Describe third platform technologies – cloud, big data, social, and mobile • Evaluate various types of intelligent storage systems and their deployments • Describe software-defined storage • Evaluate various storage networking technologies and their deployments • Articulate business continuity and archiving solutions • Describe various security threats and controls in a storage infrastructure • Describe key processes for managing storage infrastructure

Agenda

Module/Lessons		Labs

	• Introduction to Information Storage
	  • Digital Data and its Types
	  • Information Storage
	  • Key Characteristics of Data Center
	  • Evolution of Computing Platforms
	• Third Platform Technologies
	  • Cloud Computing
	  • Big Data Analytics
	  • Social Networking and Mobile computing
	  • Transforming to the Third Platform
	• Data Center Environment
	  • Data Center Infrastructure Building Blocks
	  • Compute System
	  • Compute, Application, and Desktop Virtualization
	  • Storage and Connectivity
	  • Software-Defined Data Center
	• Intelligent Storage Systems (ISS)
	  • Components of Intelligent Storage Systems
	  • RAID	

Agenda

Module/Lessons		Labs

	• Intelligent Storage Systems (ISS)
	  • Types of Intelligent Storage Systems
	• Block-based Storage System
	  • Components of Block-based Storage System
	  • Storage Provisioning
	  • Storage Tiering
	• File-based Storage System (NAS)
	  • NAS Components and Architecture
	  • File-level Virtualization and Tiering
	• Object-based and Unified Storage
	  • Object-based Storage Overview
	  • OSD Implementations and Operations
	  • Unified Storage
	• Software-Defined Storage
	  • Introduction to Software-Defined storage
	  • Control Plane
		• Storage Design
		• RAID Configuration
		• Storage Provisioning and Tiering

Agenda

Module/Lessons		Labs

	• Software-Defined Storage
	  • Software-Defined Storage Extensibility
	• Fibre Channel (FC) SAN
	  • Introduction to SAN
	  • FC SAN Overview
	  • Fibre Channel (FC) Architecture
	  • Topologies, Link Aggregation, and Zoning
	  • Virtualization in FC SAN
	• Internet Protocol (IP) SAN
	  • iSCSI
	  • Link Aggregation and VLAN
	  • FCIP
	• FC over Ethernet (FCoE) SAN
	  • Overview of FCoE SAN
	  • Converged Enhanced Ethernet (CEE)
	  • FCoE Architecture
		• FC SAN Topologies

Agenda

Module/Lessons		Labs

	• Introduction to Business Continuity
	    • Business Continuity Overview
	    • Business Continuity Solutions
	• Backup and Archive
	    • Backup and Recovery Overview
	    • Backup Method
	    • Data Deduplication
	    • Cloud-based and Mobile Device Backup
	    • Data Archiving
	• Replication
	    • Replication Overview
	    • Compute-based Replication
	    • Storage System-based Replication
	    • Network-based Replication
		• MTBF and MTTR
		• Information Availability
		• Backup and Archive

Agenda

Module/Lessons		Labs

	• Replication
	    • Data Migration and DRaaS
	• Securing the Storage Infrastructure
	    • Introduction to Information Security
	    • Storage Security Domains and Threats
	    • Security Controls
	    • Governance, Risk, and Compliance
	• Managing the Storage Infrastructure
	    • Introduction to Storage Infrastructure Management
	    • Operations Management
		• Replication

This slide intentionally left blank.

Module 1: Introduction to Information Storage

Upon completion of this module, you should be able to:

- Describe digital data, types of digital data, and information
- Describe data center and its key characteristics
- Describe key data center management processes
- Describe the evolution of computing platforms

The Growth of the Digital Universe

- The digital universe is created and defined by software
  - Digital data is continuously generated, collected, stored, and analyzed through software
- The digital universe generates approximately 4.4 trillion GB of data annually
  - Proliferation of IT, Internet usage, social media, and smart devices adds to data growth
- The Internet of Things (IoT) is also adding to data growth
  - IoT is made up of Internet-connected equipment and sensors

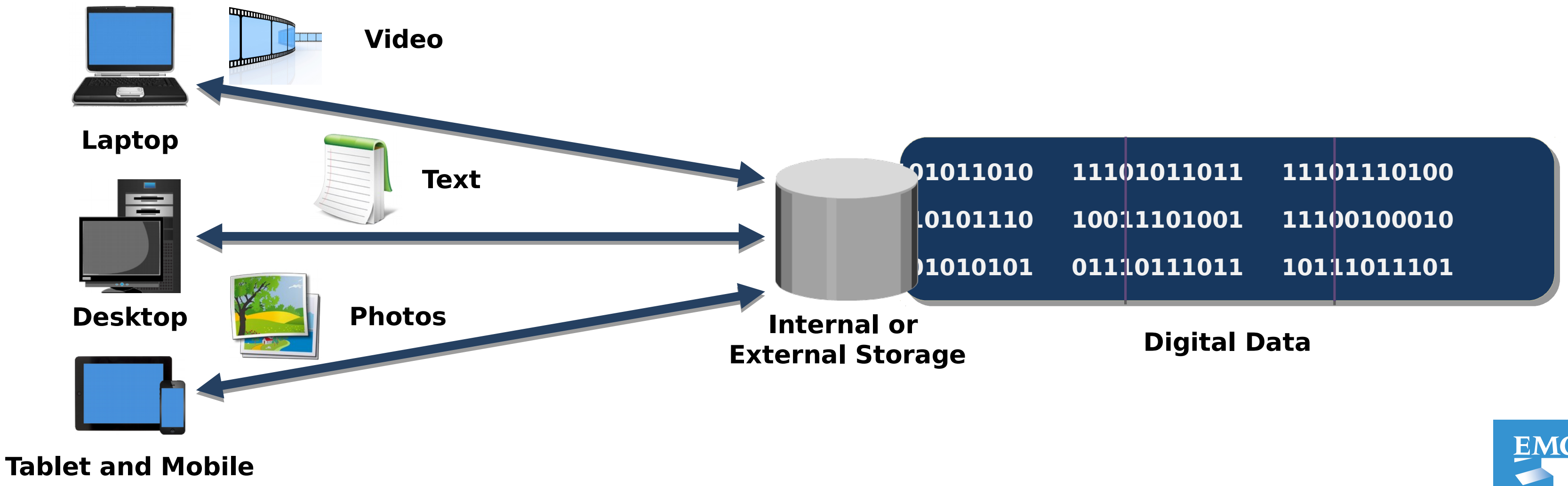
Why Information Storage and Management?

- Organizations are dependent on continuous and reliable access to information
- Organizations seek to effectively store, protect, process, manage, and leverage information
- Organizations are increasingly implementing intelligent storage solutions
  - To efficiently store and manage information
  - To gain competitive advantage
  - To derive new business opportunities

What is Digital Data?

Digital Data

A collection of facts that is transmitted and stored in electronic form, and processed through software.



Types of Digital Data

Increasing Growth

Unstructured

- Data that has no inherent structure and is usually stored as different types of files.
- E.g. Text documents, PDFs, images, and videos

Quasi-Structured

- Textual data with erratic formats that can be formatted with effort and software tools
- E.g. Clickstream data

Semi-Structured

- Textual data files with an apparent pattern, enabling analysis
- E.g. Spreadsheets and XML files

Structured

- Data having a defined data model, format, structure
- E.g. Database

What is Information?

Information

Processed data that is presented in a specific context to enable useful interpretation and decision-making.

- Example: Annual sales data processed into a sales report
  - Enables calculation of the average sales for a product and the comparison of actual sales to projected sales
- New architectures and technologies have emerged for extracting information from non-structured data

Information Storage

- Information is stored on storage devices on non-volatile media
- Types of storage devices:
  - **Magnetic storage devices**: Hard disk drive and magnetic tape
  - **Optical storage devices**: Blu-ray disc, DVD, and CD
  - **Flash-based storage devices**: Solid state drive, memory card, and USB thumb drive
- Storage devices are assembled within a storage system or “array”
  - Provides high capacity, scalability, performance, reliability, and security
- Storage systems along with other IT infrastructure are housed in a data center

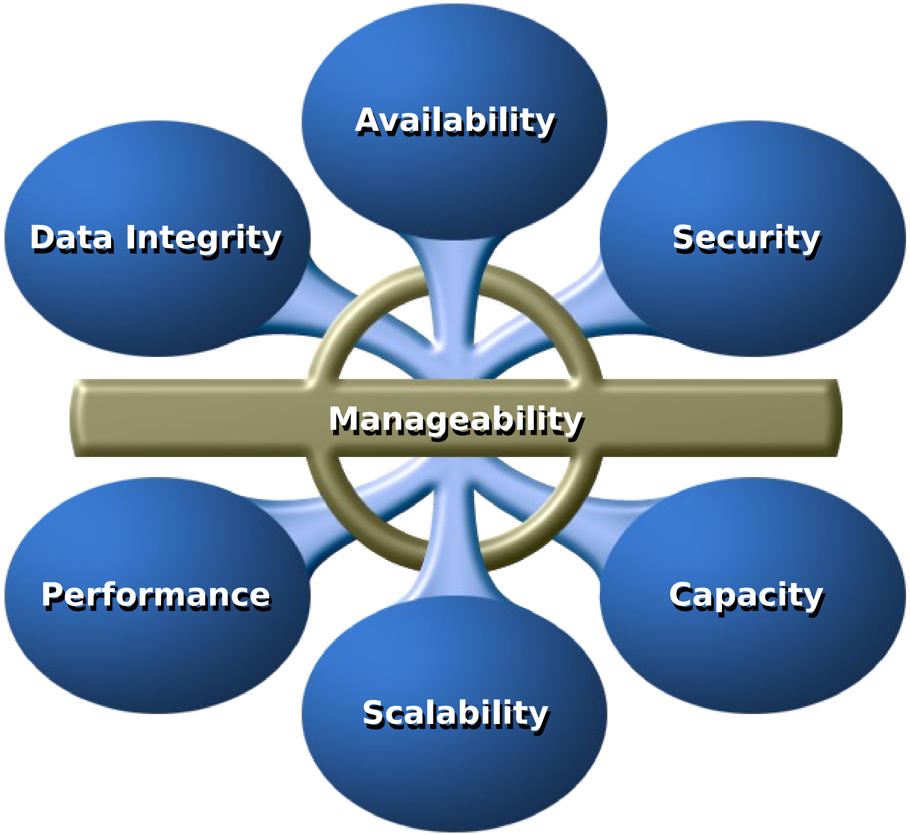
What is a Data Center?

Data Center

A facility that houses IT equipment including compute, storage, and network components, and other supporting infrastructure for providing centralized data-processing capabilities.

- A data center comprises:
  - **Facility**: The building and floor space where the data center is constructed
  - **IT equipment**: Compute, storage, and network equipment
  - **Support infrastructure**: Power supply, fire detection, HVAC, and security systems

Key Characteristics of a Data Center



Key Data Center Management Processes

Management Process	Description
Monitoring	Continuously gathering information on data center resources
Reporting	Presenting the details on resource performance, capacity, and utilization
Provisioning	Configuring and allocating resources to meet the capacity, availability, performance, and security requirements
Planning	Estimating the amount of resources required to support business operations
Maintenance	Ensuring the proper functioning of resources and resolving incidents

Evolution of Computing Platforms

PLATFORM 3

Cloud Big Data Mobile Social

Mobile Devices

BILLIONS OF USERS

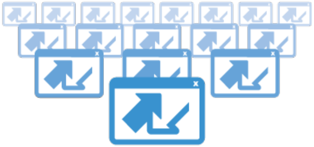
MILLIONS OF APPS

PLATFORM 2

LAN/Internet Client/Server

PC

HUNDREDS OF MILLIONS OF USERS



TENS OF THOUSANDS OF APPS

PLATFORM 1

Mainframe, Mini Computer

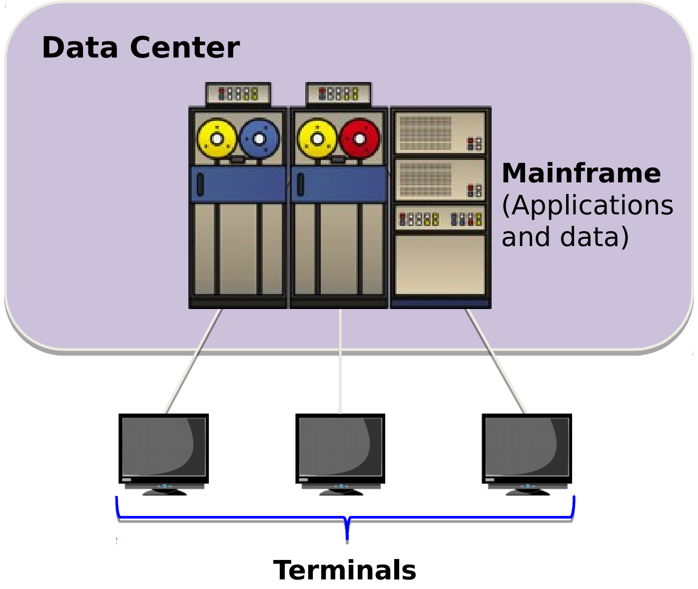
Terminals

MILLIONS OF USERS

THOUSANDS OF APPS

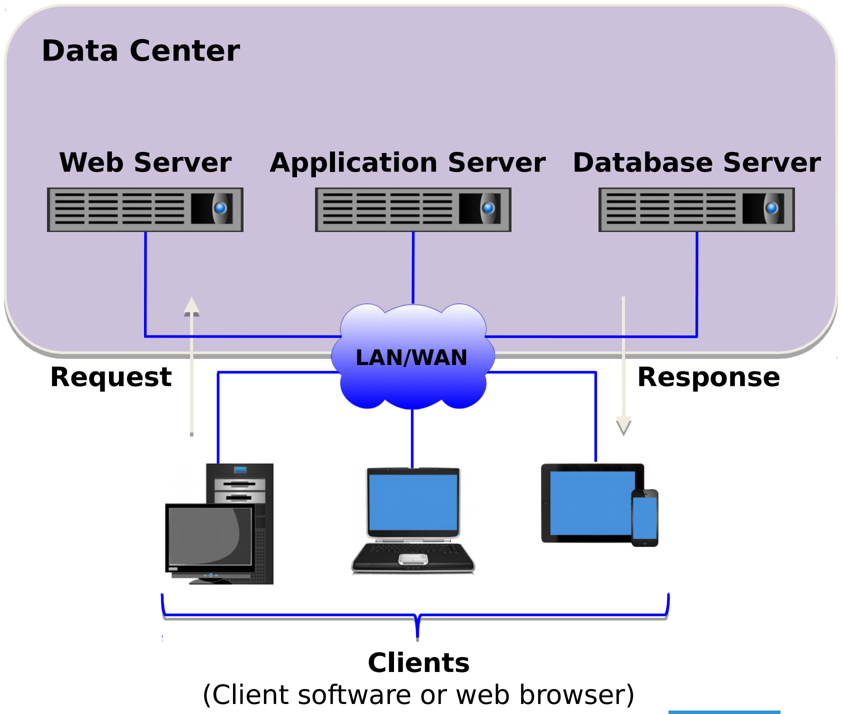
First Platform

- Based on mainframes
  - Applications and databases hosted centrally
  - Users connect to mainframes through terminals
- Challenges with mainframes
  - Substantial CAPEX and OPEX
    - High acquisition costs
    - Considerable floor space and energy requirements



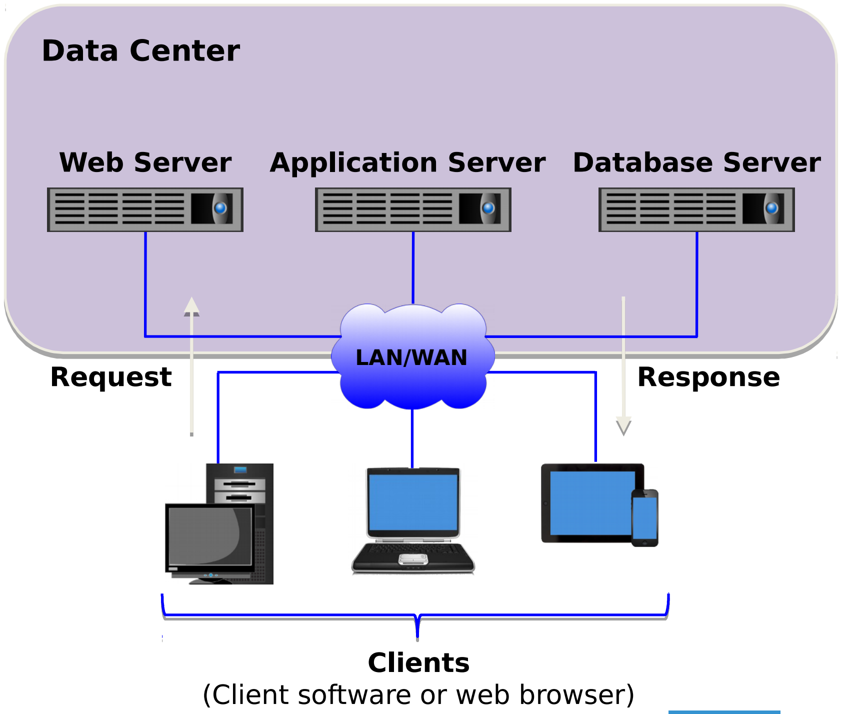
Second Platform

- Based on client-server model
  - Distributed application architecture
  - Servers receive and process requests for resources from clients
  - Users connect through a client program or a web interface
- Challenges with client-server model
  - Creation of IT silos
  - Hardware and software maintenance overhead
  - Scalability to meet the growth of users and workloads

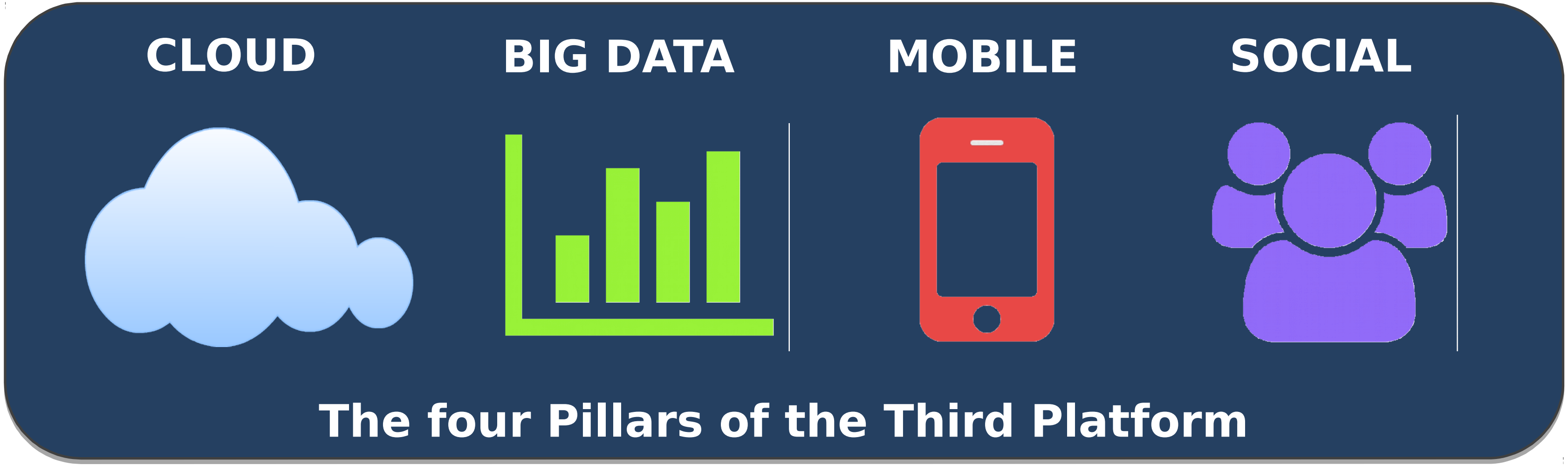


Second Platform

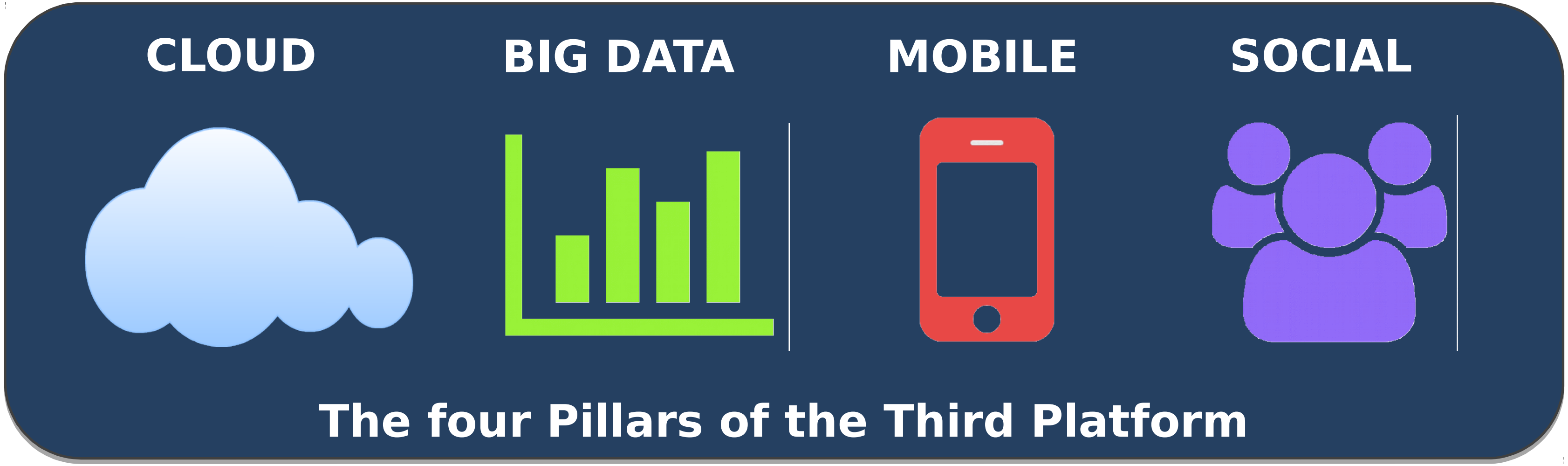
- Based on client-server model
  - Distributed application architecture
  - Servers receive and process requests for resources from clients
  - Users connect through a client program or a web interface
- Challenges with client-server model
  - Creation of IT silos
  - Hardware and software maintenance overhead
  - Scalability to meet the growth of users and workloads



Third Platform



Third Platform



Module 1: Summary

Key points covered in this module:

- Digital data, types of digital data, and information
- Data center and its key characteristics
- Key data center management processes
- Evolution of computing platforms

This slide intentionally left blank.

Module 3: Data Center Environment

Upon completion of this module, you should be able to:

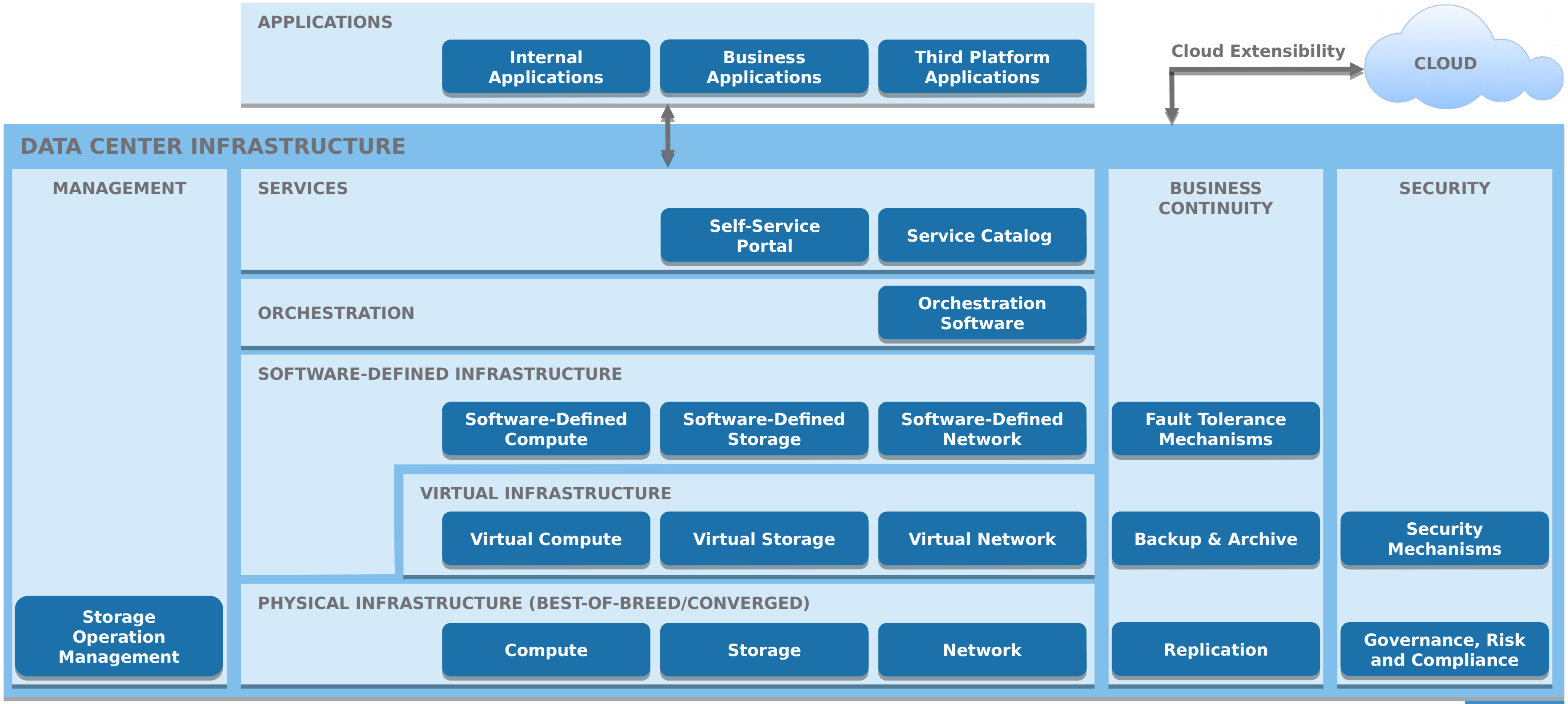
- Describe the building blocks of a data center
- Describe compute system, its components, and its types
- Describe compute virtualization, application virtualization, and desktop virtualization
- Provide an overview of storage and connectivity in a data center
- Provide an overview of software-defined data center

Lesson 1: Data Center Infrastructure Building Blocks

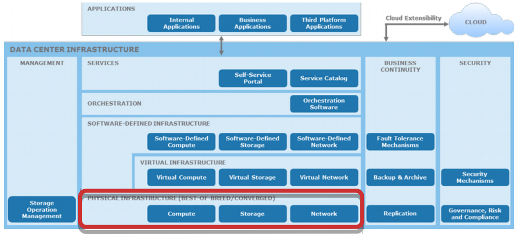
This lesson covers the following topics:

- Layers of a data center infrastructure
- Components and functions of each layer
- Cross-layer functions in a data center
- Best-of-breed vs. converged infrastructure

Data Center Infrastructure



Data Center Infrastructure

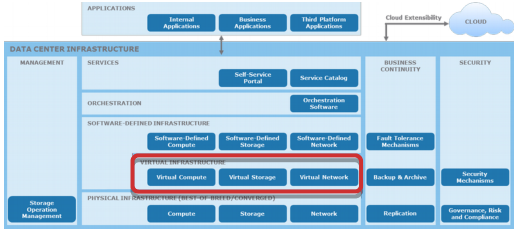


Physical Infrastructure

- Foundation layer of the data center infrastructure
- Physical components:
  - Compute systems, storage, and network devices
    - Require operating systems, system software, and protocols for their functions
- Executes the requests generated by the virtual and software-defined layers

Data Center Infrastructure

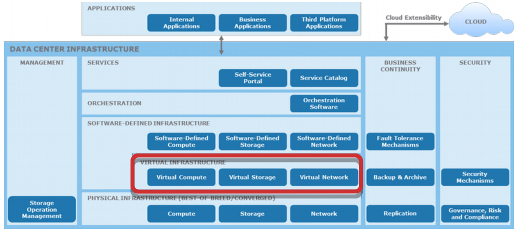
Virtual Infrastructure



- Virtualization abstracts physical resources and creates virtual resources
- Virtual components:
  - Virtual compute, virtual storage, and virtual network
    - Created from physical resource pools using virtualization software
- Benefits of virtualization:
  - Resource consolidation and multitenant environment
  - Improved resource utilization and increased ROI
  - Flexible resource provisioning and rapid elasticity

Data Center Infrastructure

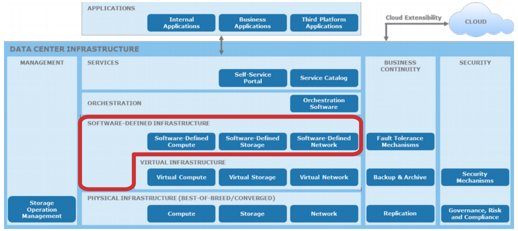
Virtual Infrastructure



- Virtualization abstracts physical resources and creates virtual resources
- Virtual components:
  - Virtual compute, virtual storage, and virtual network
    - Created from physical resource pools using virtualization software
- Benefits of virtualization:
  - Resource consolidation and multitenant environment
  - Improved resource utilization and increased ROI
  - Flexible resource provisioning and rapid elasticity

Data Center Infrastructure

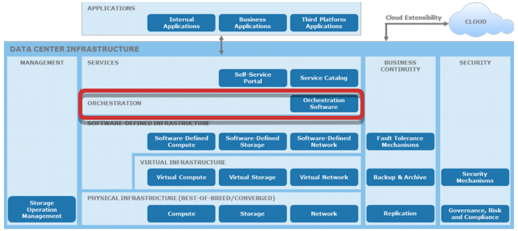
Software-defined Infrastructure



- Deployed either on virtual layer or on physical layer
- All infrastructure components are virtualized and aggregated into pools
  - Underlying resources are abstracted from applications
  - Enables ITaaS
- Centralized, automated, and policy-driven management and delivery of heterogeneous resources
- Components:
  - Software-defined compute
  - Software-defined storage
  - Software-defined network

Data Center Infrastructure

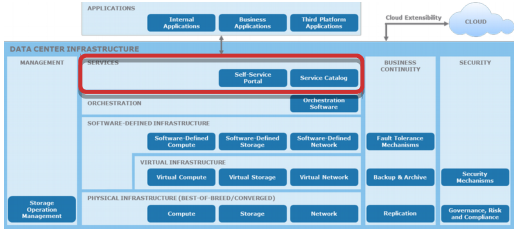
Orchestration



- Component:
  - Orchestration software
- Provides workflows for executing automated tasks
- Interacts with various components across layers and functions to invoke provisioning tasks

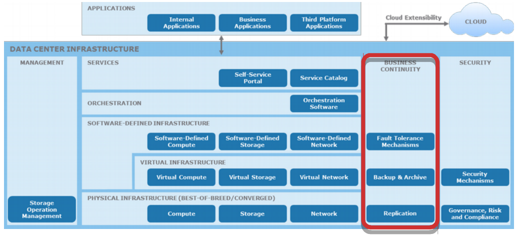
Data Center Infrastructure

Services



- Delivers IT resources as services to users
  - Enables users to achieve desired business results
  - Users have no liabilities associated with owning the resources
- Components:
  - Service catalog
  - Self-service portal
- Functions of service layer:
  - Stores service information in service catalog and presents them to the users
  - Enables users to access services via a self-service portal

Data Center Infrastructure



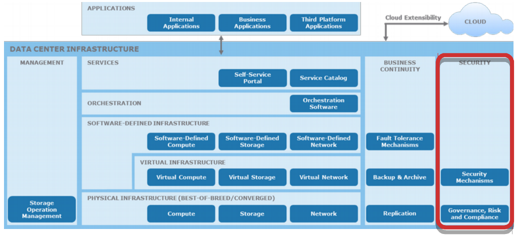
Business Continuity

- Enables ensuring the availability of services in line with SLA
- Supports all the layers to provide uninterrupted services
- Includes adoption of measures to mitigate the impact of downtime

Measure	Description
Proactive	• Business impact analysis • Risk assessment • Technology solutions deployment (backup and replication)
Reactive	• Disaster recovery • Disaster restart

Data Center Infrastructure

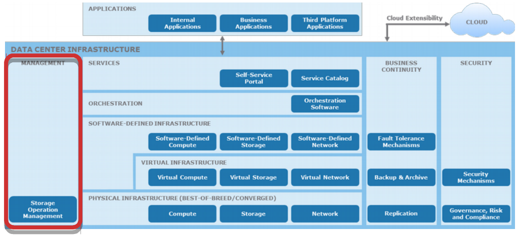
Security



- Supports all the layers to provide secure services
- Specifies the adoption of:
  - Administrative mechanisms
    - Security and personnel policies
    - Standard procedures to direct safe execution of operations
  - Technical mechanisms
    - Firewall
    - Intrusion detection and prevention systems
    - Antivirus
- Security mechanisms enable to meet governance, risk, and compliance (GRC) requirements

Data Center Infrastructure

Management



- Enables storage infrastructure configuration and capacity provisioning
- Enables problem resolution
- Enables capacity and availability management
- Enables compliance conformance
- Enables monitoring services

Best-of-breed Vs. Converged Infrastructure

Best-of-breed infrastructure	Converged infrastructure
• Integrating different best-of-breed components from multiple vendors • Prevents vendor lock-in • Enables repurposing the existing infrastructure components	• Integrates all hardware and software components into a single package • Offers a preconfigured and optimized self-contained unit • Facilitates faster acquisition and deployment

Lesson 1: Summary

During this lesson the following topics were covered:

- Layers of a data center infrastructure
- Components and functions of each layer
- Cross-layer functions of a data center
- Best-of-breed vs. converged infrastructure

Lesson 2: Compute System

This lesson covers the following topics:

- Physical and logical components of a compute system
- Types of compute systems

What is a Compute System?

- A computing platform (hardware and system software) that runs applications
  - Physical components include processor, memory, internal storage, and I/O devices
  - Logical components include OS, device drivers, file system, and logical volume manager

Physical Components of a Compute System

Processor	An IC that executes software programs by performing arithmetical, logical, and input/output operations
Random-Access Memory	Volatile data storage that contains the programs for execution and the data used by the processor
Read-Only Memory	Semiconductor memory containing boot, power management, and other device-specific firmware
Motherboard	A PCB that holds the processor, RAM, ROM, network and I/O ports, and other integrated components, such as GPU and NIC
Chipset	A collection of microchips on a motherboard to manage specific functions, such as processor access to RAM and to peripheral ports
Secondary Storage	A persistent storage device such as HDD or SSD

Physical Components of a Compute System

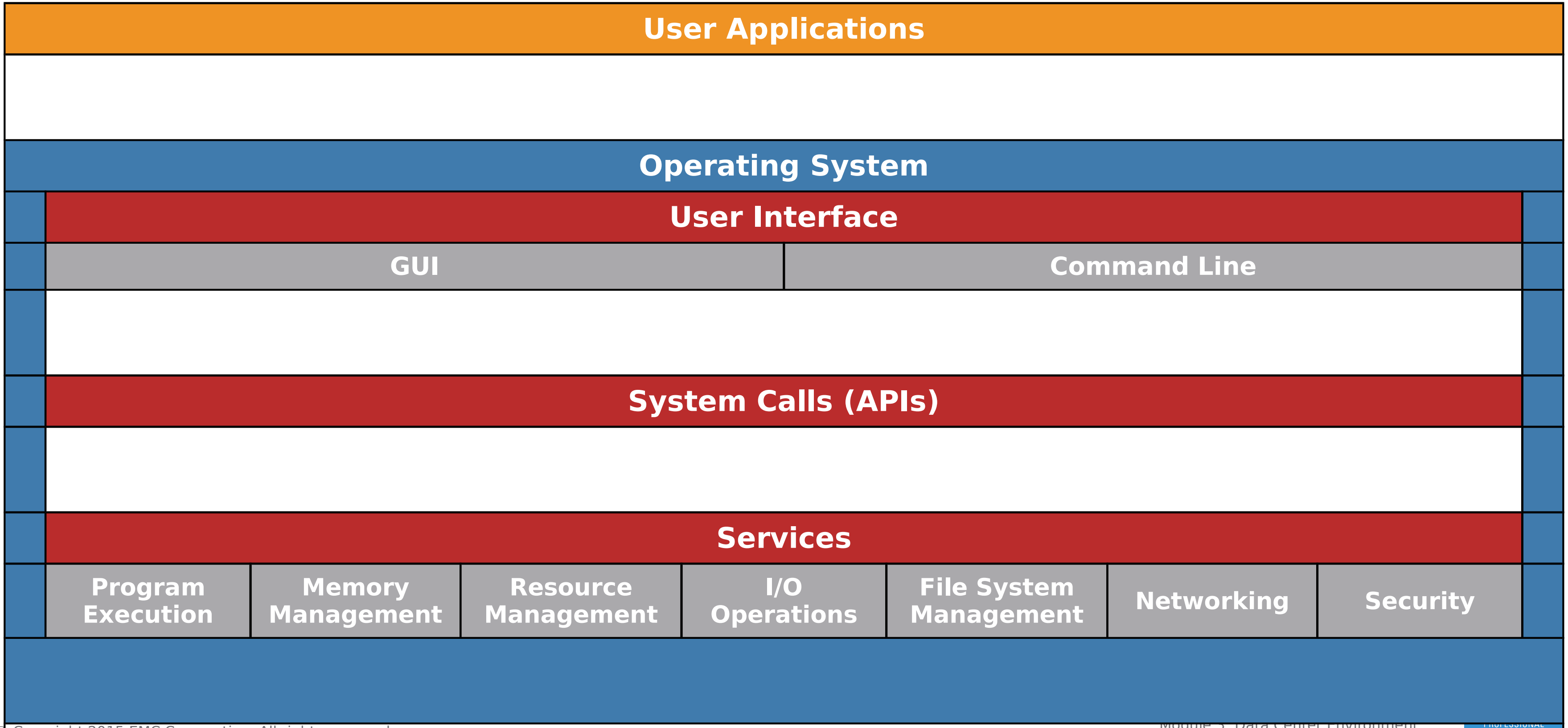
Processor	An IC that executes software programs by performing arithmetical, logical, and input/output operations
Random-Access Memory	Volatile data storage that contains the programs for execution and the data used by the processor
Read-Only Memory	Semiconductor memory containing boot, power management, and other device-specific firmware
Motherboard	A PCB that holds the processor, RAM, ROM, network and I/O ports, and other integrated components, such as GPU and NIC
Chipset	A collection of microchips on a motherboard to manage specific functions, such as processor access to RAM and to peripheral ports
Secondary Storage	A persistent storage device such as HDD or SSD

Logical Components of a Compute System

- Operating system
- Virtual memory
- Logical volume manager
- File system

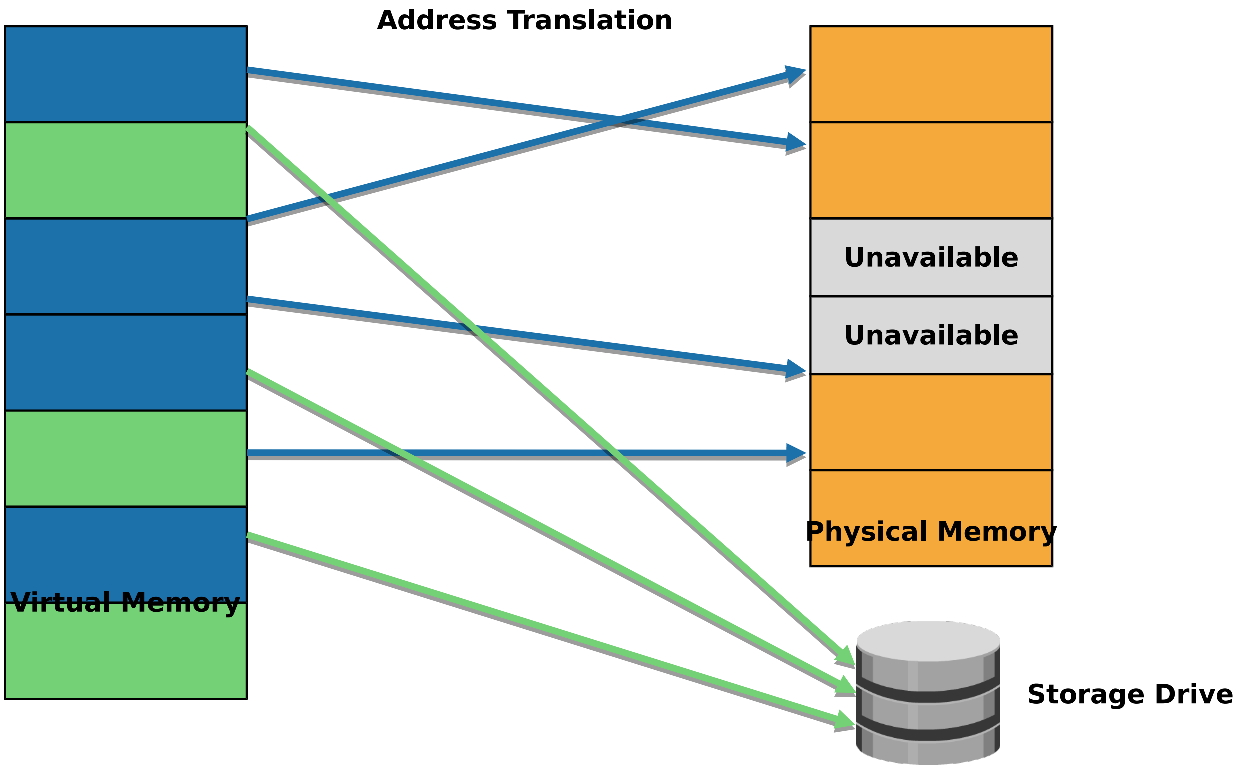
Logical Components of a Compute System

Operating System



Logical Components of a Compute System

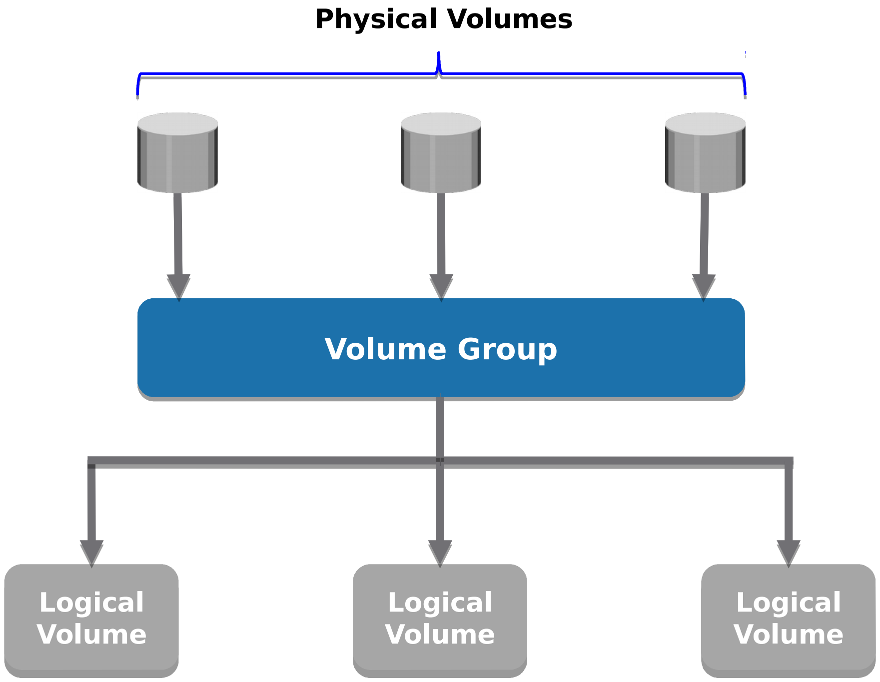
Virtual Memory



Logical Components of a Compute System

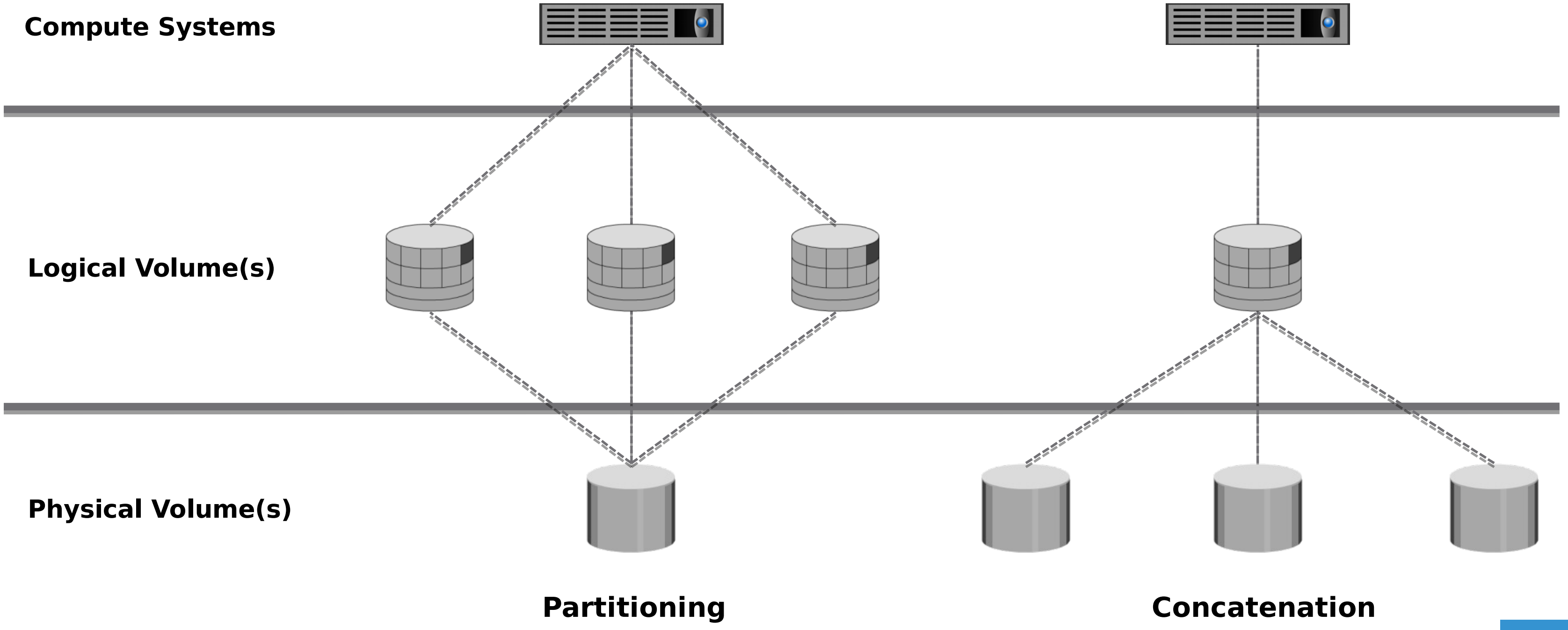
Logical Volume Manager (LVM)

- Creates and controls compute level logical storage
  - Provides a logical view of physical storage
  - Logical data blocks are mapped to physical data blocks
- Physical volumes form a volume group
  - LVM manages volume groups as a single entity
- Logical volumes are created from a volume group



Logical Components of a Compute System

LVM Example: Partitioning and Concatenation



Logical Components of a Compute System

File System

- File is a collection of related records stored as a single named unit in contiguous logical address space
- A file system controls and manages the storage and retrieval of files
  - Enables users to perform various operations on files
  - Groups and organizes files in a hierarchical structure
- File system may be broadly classified as:
  - Disk-based file system
  - Network-based file system
  - Virtual file system

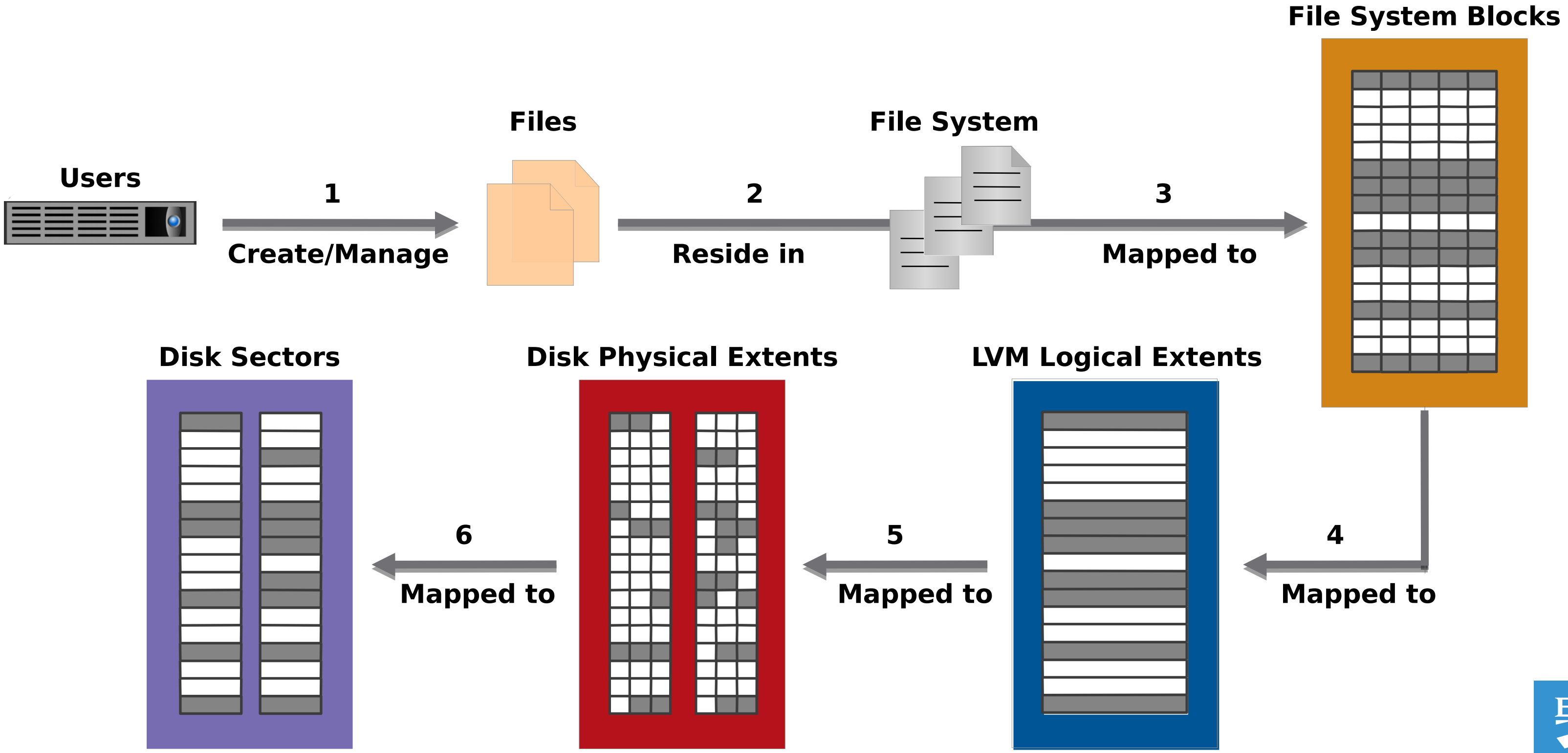
Logical Components of a Compute System

File System

- File is a collection of related records stored as a single named unit in contiguous logical address space
- A file system controls and manages the storage and retrieval of files
  - Enables users to perform various operations on files
  - Groups and organizes files in a hierarchical structure
- File system may be broadly classified as:
  - Disk-based file system
  - Network-based file system
  - Virtual file system

Logical Components of a Compute System

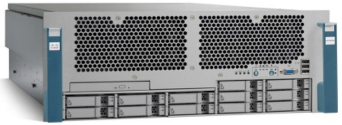
File System (Cont'd)



Types of Compute Systems

Tower Compute System	Rack-mounted Compute System	Blade Compute System
		

Types of Compute Systems

Tower Compute System	Rack-mounted Compute System	Blade Compute System
	 	 

Lesson 2: Summary

During this lesson the following topics were covered:

- Physical and logical components of a compute system
- Types of compute systems

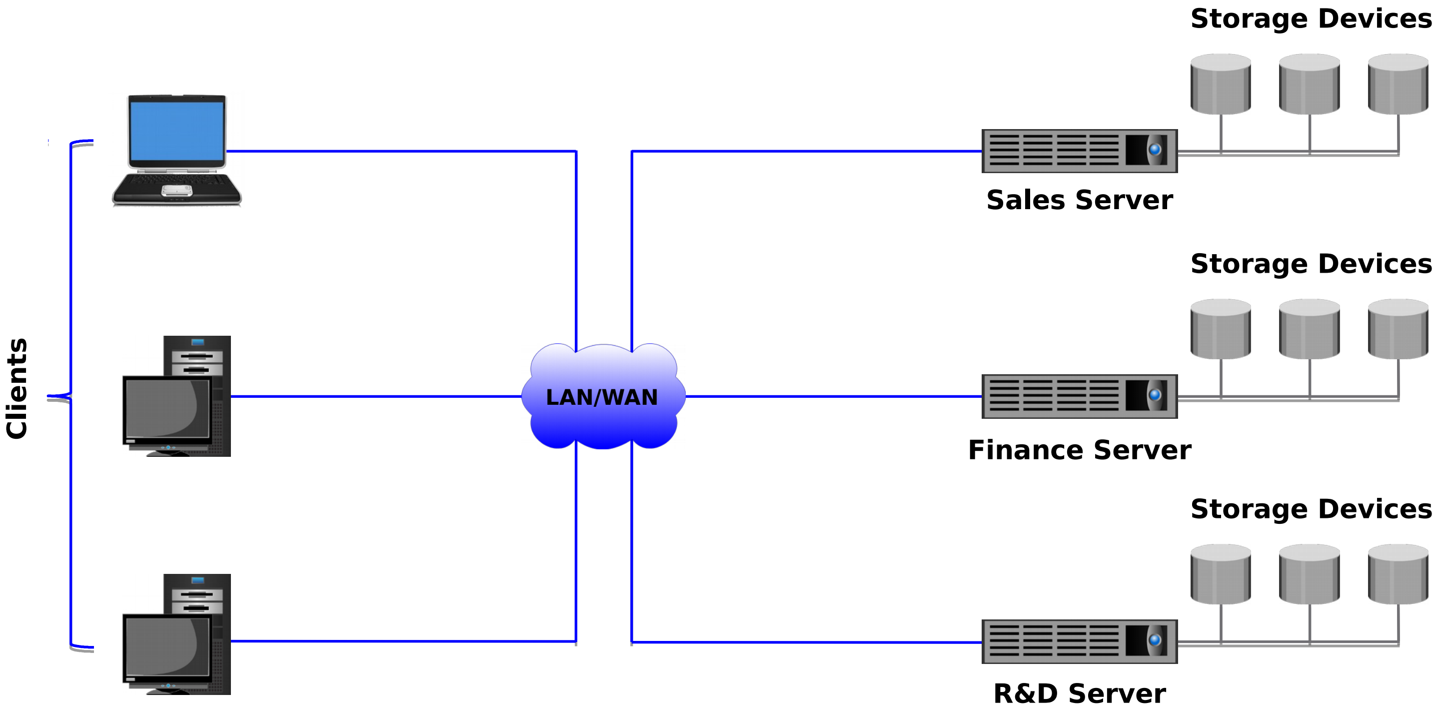
Lesson 4: Storage and Connectivity

This lesson covers the following topics:

- Evolution of storage architecture
- Types of storage devices
- Compute-to-compute and compute-to-storage connectivity
- Storage connectivity protocols

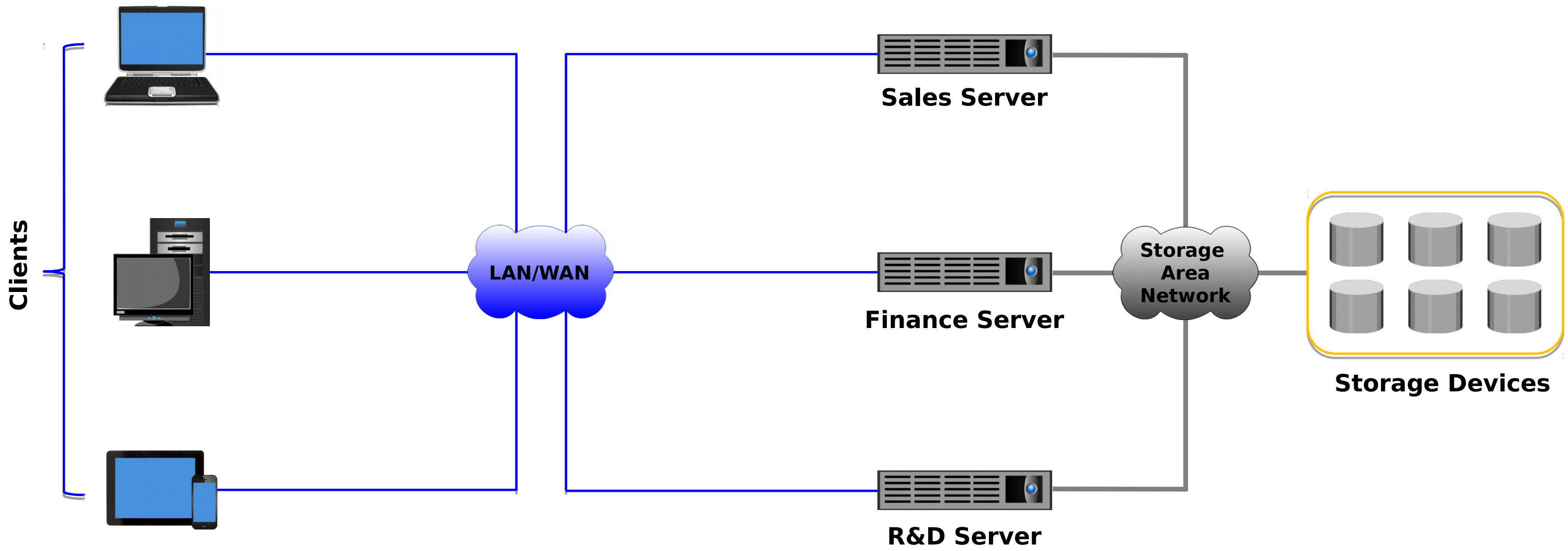
Evolution of Storage Architecture

Server-centric Storage Architecture



Evolution of Storage Architecture (Cont'd)

Information-centric Storage Architecture



Types of Storage Devices

Magnetic disk drive	• Stores data on a circular disk with a ferromagnetic coating • Provides random read/write access • Most popular storage device with large storage capacity
Solid-state (flash) drive	• Stores data on a semiconductor-based memory • Very low latency per I/O, low power requirements, and very high throughput
Magnetic tape drive	• Stores data on a thin plastic film with a magnetic coating • Provides only sequential data access • Low-cost solution for long term data storage
Optical disc drive	• Stores data on a polycarbonate disc with a reflective coating • Write Once and Read Many capability: CD, DVD, BD • Low-cost solution for long-term data storage

Types of Storage Devices

Magnetic disk drive	• Stores data on a circular disk with a ferromagnetic coating • Provides random read/write access • Most popular storage device with large storage capacity
Solid-state (flash) drive	• Stores data on a semiconductor-based memory • Very low latency per I/O, low power requirements, and very high throughput
Magnetic tape drive	• Stores data on a thin plastic film with a magnetic coating • Provides only sequential data access • Low-cost solution for long term data storage
Optical disc drive	• Stores data on a polycarbonate disc with a reflective coating • Write Once and Read Many capability: CD, DVD, BD • Low-cost solution for long-term data storage

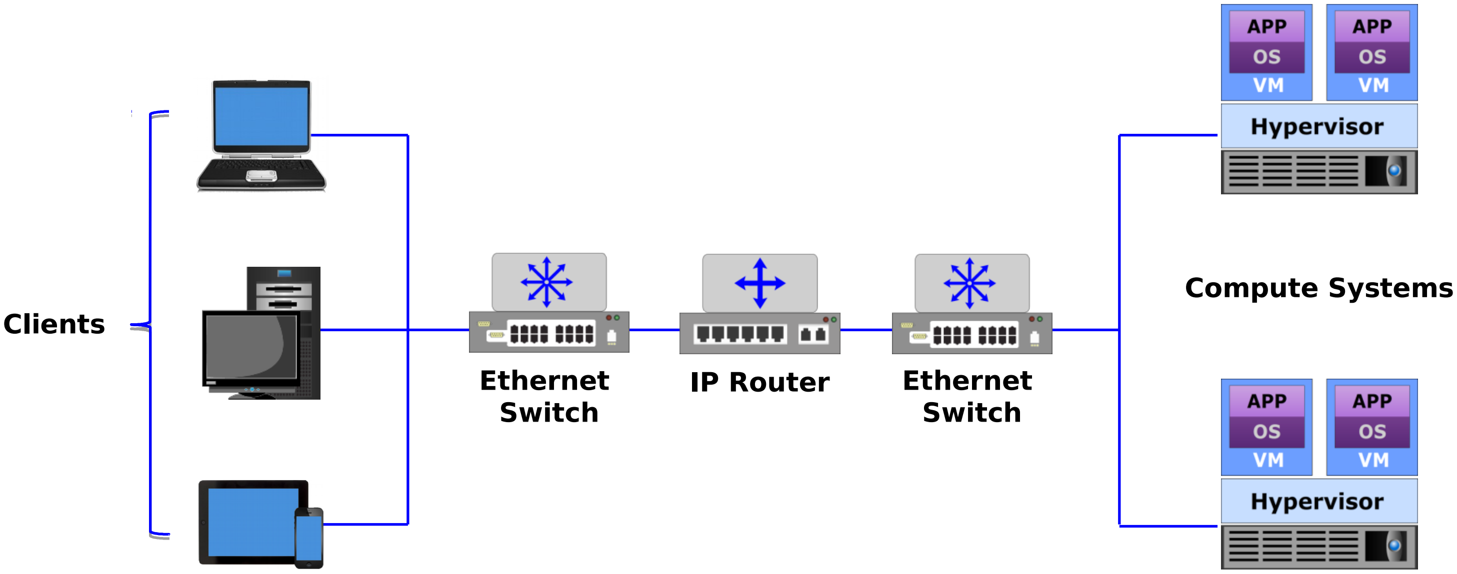
Overview of Storage Virtualization

- Abstracts physical storage resources to create virtual storage resources:
  - Virtual volumes
  - Virtual disk files
  - Virtual storage systems
- Storage virtualization software can be:
  - Built into the operating environment of a storage system
  - Installed on an independent compute system
  - Built into a hypervisor

Introduction to Connectivity

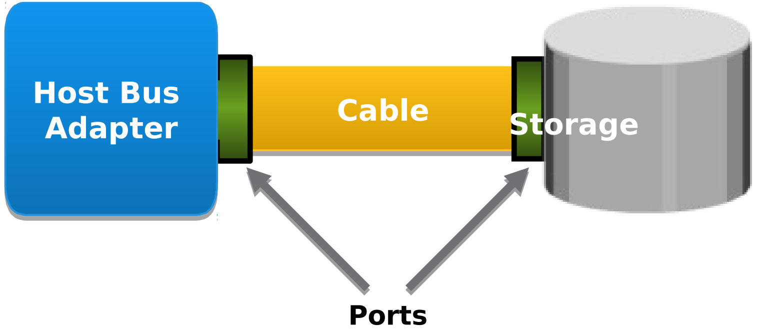
- Communication paths between IT infrastructure components for information exchange and resource sharing
- Types of connectivity:
  - Compute-to-compute connectivity
  - Compute-to-storage connectivity

Compute-to-compute Connectivity



Compute-to-storage Connectivity

- Enabled through physical components and interface protocols
- Physical connectivity components:
  - Host bus adapter, port, and cable
- Protocols define formats for communication between devices
  - Popular storage interface protocols are IDE/ATA, SCSI, and FC
- Storage may be connected directly or over a SAN



Storage Connectivity Protocols

Protocol	Description
IDE/ATA	• Popular interface used to connect hard disks and optical drives • The Ultra DMA/133 version of ATA supports a throughput of 133 MB/s
Serial ATA	• Serial version of the IDE/ATA specification typically used for internal connectivity • Provides data transfer rate up to 16 Gb/s (standard 3.2)
SCSI	• Popular standard for compute-to-storage connectivity • Supports up to 16 devices on a single bus • Ultra-640 version provides data transfer speed up to 640 MB/s
SAS	• Point-to-point serial protocol replacing parallel SCSI • Supports data transfer rate up to 12 Gb/s (SAS 3.0)
FC	• Widely-used protocol for high speed compute-to-storage communication • Provides a serial data transmission that operates over copper wire and/or optical fiber • Latest version of the FC interface ‘16FC’ allows transmission of data up to 16 Gb/s
IP	• Existing IP-based network leveraged for storage communication • Examples: iSCSI and FCIP protocols

Storage Connectivity Protocols

Protocol	Description
IDE/ATA	• Popular interface used to connect hard disks and optical drives • The Ultra DMA/133 version of ATA supports a throughput of 133 MB/s.
Serial ATA	• Serial version of the IDE/ATA specification typically used for internal connectivity • Provides data transfer rate up to 16 Gb/s (standard 3.2)
SCSI	• Popular standard for compute-to-storage connectivity • Supports up to 16 devices on a single bus • Ultra-640 version provides data transfer speed up to 640 MB/s
SAS	• Point-to-point serial protocol replacing parallel SCSI • Supports data transfer rate up to 12 Gb/s (SAS 3.0)
FC	• Widely-used protocol for high speed compute-to-storage communication • Provides a serial data transmission that operates over copper wire and/or optical fiber • Latest version of the FC interface ‘16FC’ allows transmission of data up to 16 Gb/s
IP	• Existing IP-based network leveraged for storage communication • Examples: iSCSI and FCIP protocols

Overview of Network Virtualization

- Abstracts physical network resources to create virtual network resources:
  - Virtual switch
  - Virtual LAN
  - Virtual SAN
- Network virtualization software can be:
  - Built into the operating environment of a network device
  - Installed on an independent compute system
  - Built into a hypervisor

Lesson 4: Summary

During this lesson the following topics were covered:

- Evolution of storage architecture
- Types of storage devices
- Compute-to-compute and compute-to-storage connectivity
- Storage connectivity protocols

Welcome to Information Storage and Management Version 3.

Information Storage and Management (ISM) is a unique course that provides a comprehensive understanding of the various storage infrastructure components in data center environments. It enables participants to make informed decisions on storage-related technologies in an increasingly complex IT environment, which is fast changing with the adoption of software-defined infrastructure management and third platform technologies (cloud, Big Data, social, and mobile technologies). It provides a strong understanding of storage technologies and prepares participants for advanced concepts, technologies, and processes. Participants will learn the architectures, features, and benefits of intelligent storage systems including block-based, file-based, object-based, and unified storage; software-defined storage; storage networking technologies such as FC SAN, IP SAN, and FCoE SAN; business continuity solutions such as

This module focuses on digital data, the types of digital data, and information. This module also focuses on data center and its key characteristics. Further this module focuses on the key data center management processes. Finally, this module focuses on the evolution of computing platforms.

We live in a *digital universe* – a world that is created and defined by software. A massive amount of digital data is continuously generated, collected, stored, and analyzed through software in the digital universe. According to the 2014 Digital Universe Study conducted by International Data Corporation (IDC), it is estimated that the digital universe produces approximately 4.4 trillion gigabytes (GB) of data annually, which is doubling every two years. By these estimates, it is projected that by the year 2020, the digital universe will expand to 44 trillion GB of data. The data in the digital universe comes from diverse sources, including individuals living and working online, organizations employing information technology (IT) to run their businesses, and from a variety of "smart" electronic devices connected to the Internet.

In organizations, the volume and importance of information for business operations continue to grow at astounding rates. Individuals constantly generate and consume information through numerous activities, such as web searches, e-mails, uploading and downloading content and sharing media files. The rapid proliferation of online social networking and Internet-enabled smartphones and tablets has also contributed significantly to the growth of the digital universe.

The advent of the *Internet of Things* (IoT) is also gradually adding to the growth of the digital universe. The IoT is a technology trend wherein "smart" devices with embedded electronics, software, and sensors exchange data with other devices over the Internet. Examples of such devices are wearable gadgets – smartwatches and fitness activity trackers; electronic sensors – temperature sensors and heart monitoring implants; and household appliances – televisions, thermostats, and lighting. The IoT has vast applications and is driving the development of several innovative technology solutions. Some application areas include weather monitoring – remote monitoring and analysis of temperature and atmospheric conditions; healthcare – health monitoring devices can enable doctors to remotely monitor patients and be notified in case of emergencies; and infrastructure management – technicians can remotely monitor equipment and proactively schedule repair activities for maintenance crews.

Organizations have become increasingly information-dependent in the twenty-first century, and information must be available whenever and wherever it is required. It is critical for users and applications to have continuous, fast, reliable, and secure access to information for business operations to run as required. Some examples of such organizations and processes include banking and financial institutions, government departments, online retailers, airline reservations, billing and transaction processing, social networks, stock trading, scientific research, and healthcare.

It is essential for organizations to store, protect, process, and manage information in an efficient and cost-effective manner. Legal, regulatory, and contractual obligations regarding the availability, retention, and protection of data further add to the challenges of storing and managing information.

Organizations also face newer challenges in the form of requirement to extract value from the information generated in the digital universe. Information can be leveraged to identify opportunities to transform and enhance businesses and gain a competitive edge. For example, an online retailer may need to identify the preferred product types and brands of customers by analyzing their search, browsing, and purchase patterns. The retailer can then maintain a sufficient inventory of popular products, and also advertise relevant products to the existing and potential customers. Furthermore, the IoT is expected to lead to new consumer and business behavior in the coming years creating new business opportunities.

To meet all these requirements and more, organizations are increasingly undertaking digital transformation initiatives to implement intelligent storage solutions. These solutions not only enable efficient and optimized storage and management of information, but also enable extraction of value from information to derive new business opportunities, gain a competitive advantage, and create new sources of revenue.

A generic definition of *data* is that it is a collection of facts, typically collected for the purpose of analysis or reference. Data can exist in a variety of forms such as facts stored in a person's mind, photographs and drawings, alphanumeric text and images in a book, a bank ledger, and tabled results of a scientific survey. Originally, data is the plural form of “datum”. However, data is now generally treated as a singular or mass noun representing a collection of facts and figures. This is especially true when referring to digital data.

In computing, *digital data* is a collection of facts that is transmitted and stored in electronic form, and processed through software. Digital data is generated by various devices, such as desktops, laptops, tablets, mobile phones, and electronic sensors. It is stored as strings of binary values (0s and 1s) on a storage medium that is either internal or external to the devices generating or accessing

Based on how it is stored and managed, digital data can be broadly classified as either structured data or unstructured data. *Structured data* is organized in fixed fields within a record or file. For data to be structured, a data model is required. A *data model* specifies the format for organizing data, and also specifies how different data elements are related to each other. For example, in a relational database, data is organized in rows and columns within named tables. *Semi-structured data* does not have a formal data model but has an apparent, self-describing pattern and structure that enable its analysis. Examples of semi-structured data include spreadsheets that have a row and column structure, and XML files that are defined by an XML schema. *Quasi-structured data* consists of textual data with erratic data formats, and can be formatted with effort, software tools, and time. An example of quasi-structured data is a “clickstream” or “clickpath” that includes data about which webpages a user visited and in what order – which is the result of the successive mouse clicks the user made. A clickstream shows when a user entered a website, the pages viewed, the time spent on each page, and when the user exited. *Unstructured data* does not have a data model and is not organized in any particular format. Some examples of unstructured data include text documents, PDF files, e-mails, presentations, images, and videos.

As indicated by the figure on the slide, the majority, which is more than 90 percent, of the data generated in the digital universe today is *non-structured data* (semi-, quasi-, and unstructured). Although the figure shows four different and separate types of data, in reality a mixture of these is typically generated. For instance, in a call center for customer support of a software product, a classic relational database management system (RDBMS) may store call logs with structured data such as date/time stamps, machine types, and problem type entered by the support desk person. In addition, there may be unstructured or semi-structured data, such as an e-mail ticket of the problem, call log information, or the actual call recording.

The terms “data” and “information” are closely related and it is common for the two to be used interchangeably. However, it is important to understand the difference between the two. Data, by itself, is simply a collection of facts that needs to be processed for it to be useful. For example a set of annual sales figures of an organization is data. When data is processed and presented in a specific context it can be interpreted in a useful manner. This processed and organized data is called *information*. For example, when the annual sales data is processed into a sales report, it provides useful information, such as the average sales for a product (indicating product demand and popularity), and a comparison of the actual sales to the projected sales. Information thus creates knowledge and enables decision-making.

As discussed previously, processing and analyzing data is vital to any organization. It enables

In a computing environment, storage devices (or simply “storage”) are devices consisting of non-volatile recording media on which information can be persistently stored. Storage may be internal (for example, internal hard drive), removable (for example, memory cards), or external (for example, magnetic tape drive) to a compute system. Based on the nature of the storage media used, storage devices can be broadly classified as given below:

- **Magnetic storage devices**: For example, hard disk drive and magnetic tape drive.
- **Optical storage devices**: For example, Blu-ray, DVD, and CD.
- **Flash-based storage devices**: For example, solid state drive (SSD), memory card, and USB thumb drive (or pen drive).

Storage is a core component in an organization’s IT infrastructure. Various factors such as the media, architecture, capacity, addressing, reliability, and performance influence the choice and use of storage devices in an enterprise environment. For example, disk drives and SSDs are used for storing business-critical information that needs to be continuously accessible to applications; whereas, magnetic tapes and optical storage are typically used for backing up and archiving data. The different types of storage devices are covered in Module 3, ‘Data Center Environment’.

In enterprise environments, information is typically stored on storage systems (or storage “arrays”). A storage system is a hardware component that contains a group of homogeneous/heterogeneous storage devices assembled within a cabinet. These enterprise-class storage systems are designed for high capacity, scalability, performance, reliability, and security to meet business requirements. The compute systems that run business applications are provided storage capacity from storage systems. Storage systems are covered in Module 4, ‘Intelligent Storage Systems (ISS)’. Organizations typically house their IT infrastructure, including compute systems, storage systems, and network equipment within a data center.

A data center is a dedicated facility where an organization houses, operates, and maintains back-end IT infrastructure including compute systems, storage systems, and network equipment along with other supporting infrastructure. A data center centralizes an organization's IT equipment and data-processing operations, and is vital for carrying out business operations.

A data center typically comprises the following:

- **Facility:** It is the building and floor space where the data center is constructed. It typically has a raised floor with ducts underneath holding power and network cables.
- **IT equipment:** It includes equipment such as compute systems, storage systems, network equipment and cables, and cabinets for housing the IT equipment.
- **Support infrastructure:** It includes all the equipment necessary to securely sustain the functioning of the data center. Some key support equipment are power equipment including uninterruptible power sources, and power generators; environmental control equipment including fire and water detection systems, heating, ventilation, and air conditioning (HVAC) systems; and security systems including biometrics, keycard, and video surveillance systems.

An organization may build a data center to provide open access to applications over the Internet, or for privately executing business applications within its operational environment. A data center may be constructed in-house and located in an organization's own facility, or it may be outsourced, with equipment being located at a third-party site. Large organizations often maintain multiple data centers to distribute data-processing workloads and for disaster recovery.

Organizations are increasingly focusing on energy-efficient technologies and efficient management practices to reduce the energy consumption of data centers and lessen the impact on the environment. Such data centers are called as "green data centers".

Data centers are designed and built to fulfill the key characteristics shown in the figure on the slide. Although the characteristics are applicable to almost all data center components, the discussion here primarily focuses on storage systems.

Availability: Availability of information as and when required should be ensured. Unavailability of information can severely affect business operations, lead to substantial financial losses, and damage the reputation of an organization.

Security: Policies and procedures should be established, and control measures should be implemented to prevent unauthorized access to and alteration of information.

Capacity: Data center operations require adequate resources to efficiently store and process large and increasing amounts of data. When capacity requirements increase, additional capacity should be provided either without interrupting the availability or with minimal disruption. Capacity may be managed by adding new resources or by reallocating existing resources.

Scalability: Organizations may need to deploy additional resources such as compute systems, new applications, and databases to meet the growing requirements. Data center resources should scale to meet the changing requirements, without interrupting business operations.

Performance: Data center components should provide optimal performance based on the required service levels.

Data integrity: Data integrity refers to mechanisms, such as error correction codes or parity bits, which ensure that data is stored and retrieved exactly as it was received.

Manageability: A data center should provide easy, flexible, and integrated management of all its components. Efficient manageability can be achieved through automation for reducing manual intervention in common, repeatable tasks.

The activities carried out to ensure the efficient functioning of a data center can be broadly categorized under the following key management processes:

Monitoring: It is a continuous process of gathering information on various resources in the data center. The process involves monitoring parameters such as configuration, availability, capacity, performance, and security of resources.

Reporting: It is a process of collating and presenting the monitored parameters such as resource performance, capacity, and utilization of resources. Reporting enables data center managers to analyze and improve the utilization of data center resources and identify problems. It also helps in establishing business justifications and chargeback of costs associated with data center operations.

Provisioning: It is the process of configuring and allocating the resources that are required to carry out business operations. For example, compute systems are provisioned to run applications and storage capacity is provisioned to a compute system. Provisioning primarily includes resource management activities to meet capacity, availability, performance, and security requirements.

Planning: It is a process of estimating the amount of IT resources required to support business operations and meet the changing resource requirements. Planning leverages the data collected during monitoring and enables improving the overall utilization and performance of resources. It also enables estimation of future resource requirements. Data center managers also determine the impact of incidents and devise contingency plans to resolve them.

Maintenance: It is a set of standard repeatable activities for operating the data center. It involves ensuring the proper functioning of resources and resolving incidents such as malfunctions, outages, and equipment loss. It also involves handling identified problems or issues within the data center and incorporating changes to prevent future problem occurrence.

In general, the term “platform” refers to hardware and software that are associated with a particular computing architecture deployed in a data center. Computing platforms evolve and grow with advances and changes in technology. The figure on the slide displays the three computing platforms of IT growth as specified by IDC. The *first platform* (or Platform 1) dates back to the dawn of computing and was primarily based on mainframes and terminals. The *second platform* (or Platform 2) emerged with the birth of the personal computer (PC) in the 1980s and was defined by the client-server model, Ethernet, RDBMSs, and web applications. The *third platform* (or Platform 3) of today comprises cloud, Big Data, mobile, and social technologies.

Each computing platform is defined not so much by the comprising technologies but by the scale of users and the scope of applications the

Mainframes are compute systems with very large processing power, memory, and storage capacity and are primarily used for centrally hosting mission-critical applications and databases in an organization's data center. Multiple users simultaneously connect to mainframes through less-powerful devices, such as workstations or terminals. All processing is performed on the mainframe, while the terminals only provide an interface to use the applications and view results. Although mainframes offer high reliability and security, there are several cost concerns associated with them. Mainframes have high acquisition costs, and considerable floor space and energy requirements. Deploying mainframes in a data center may involve substantial capital expense (CAPEX) and operating expense (OPEX). Historically, large organizations such as banks, insurance agencies, and government

The *client-server model* uses a distributed application architecture, in which a compute system called “server” runs a program that provides services over a network to other programs running on various end-point devices called “clients”. Server programs receive requests for resources from client programs and in response to the requests, the clients receive access to resources, such as e-mail applications, business applications, web applications, databases, files, and printers. Client devices can be desktops, laptops, and mobile devices. Clients typically communicate with servers over a LAN or WAN, with users making use of either a client application or a web interface on a browser.

In the client-server model, both the clients and the servers may have distinct processing tasks that they routinely perform. For example, a client may run the business application while the server may run the database management system (DBMS) to manage storage and retrieval of information to and from a database. This is called a *two-tier architecture*. Alternatively, a client may use an application or web interface to accept information while the server runs another application that processes the information and sends the data to a second server that runs the DBMS. This is called the *three-tier architecture*. This distributed application architecture can be extended to any number of tiers (*n-tier architecture*). Because both client and server systems are intelligent devices, the client-server model is completely different from the mainframe model.

The figure on the slide shows an example of the client-server model. In the example, clients interact with the web server using a web browser. The web server processes client requests through HTTP and delivers HTML pages. The application server hosts a business application and the database server hosts a DBMS. The clients interact with the application server through client software. The application server communicates with the database server to retrieve information and provide results to the clients. In some implementations, applications and databases may even be hosted on the same server.

(Cont’d)

Some challenges with the client-server model are associated with creation of IT silos, maintenance overhead, and scalability issues. In organizations, it is common for business units/departments to have their own servers running business applications. This leads to the creation of application and information silos (individual, disparate systems). Silos make it difficult to efficiently utilize or share IT resources, and are challenging to manage and integrate. Though the cost of server hardware is considerably less than mainframes, there is still a significant OPEX involved in maintenance of multiple servers and clients, and the software running on them. Furthermore, in this model, it is challenging to meet today's rapid growth in users, information, and applications workloads. Adding more servers does not necessarily lead to better workload management. It is also necessary to optimally distribute processing and application logic across servers and application instances.

Note: In general, a compute system is a device with an operating system (OS) that runs applications. Physical servers, hosts, desktops, laptops, and mobile devices are examples of compute systems. In this course, the term compute system or compute is used to refer to physical servers and hosts on which business applications of an organization are deployed.

The term “third platform” was coined by IDC, and Gartner refers to the same as a “nexus of forces”. The third platform is built on a foundation of cloud, Big Data, mobile, and social technologies. These are the four major “disruptive” technologies that are significantly transforming businesses, economies, and lives globally.

At its core, the third platform has the cloud that enables a consumer to provision IT resources as a service from a cloud provider. Big Data enables analytics that create deeper insights from data for improved decision-making. Mobile devices enable pervasive access to applications and information. Social technologies connect individuals, and enable collaboration and information exchange.

Over the past three decades, it was essential for organizations to intelligently leverage the second platform for their businesses. According to IDC, over the next three decades, the third platform will represent the basis for solution development and business innovation. The third platform is being used for the digital transformation, evolution, and expansion of all industries and for developing major new sources of competitive advantage. Business strategists, IT leaders, and solution developers are already building disruptive new business models and consumer services around third platform technologies.

Third platform technologies are an enhancement of second platform technologies rather than a substitution. A key aspect of third platform is that it is a convergence of cloud, Big Data, mobile, and social technologies and not just each technology taken in isolation. The real key is combining two or more of the technologies to create high-value industry solutions known as “*mashups*”. For example, some of the top drivers of cloud include social and mobile solutions. This means that organizations already see the greatest value in solutions that are mashups across all four technologies. The combinations of third platform technologies are already transforming organizations such as retail, financial services, government departments, telecommunications, and healthcare.

(Cont’d)

- According to IDC, it is estimated that currently over 80 percent of the infrastructure and applications in most data centers belong to the second platform. Second platform technologies also currently account for 74 percent of worldwide IT spending. This means that for organizations that have a significant investment in second platform technologies, an immediate and complete shift to the third platform may not be cost-effective and practical. This has led to an intermediate computing platform called "Platform 2.5", between the second and third platforms. *Platform 2.5* includes the solutions and technologies that enable organizations to bridge the gap between the second and third platforms. Platform 2.5 technologies enable organizations to use a combination of second and third platform technologies. Organizations would be able to deliver second platform applications and build third platform outcomes without duplicating and moving data. For example, platform 2.5 technologies would allow an organization to run second platform applications using traditional data structures and protocols, while enabling the same data to be leveraged for analytics using Big Data technologies.
- IDC predicts that future global IT spending will primarily focus on segments such as wireless data, smartphones and tablets, cloud services, Big Data analytics, and IoT. This spending is estimated

This module covered digital data, the types of digital data, and information. This module also covered data center and its key characteristics. Further, this module covered the key data center management processes. Finally, this module covered the evolution of computing platforms.

Module 3: Data Center Environment

Upon completion of this module, you should be able to:

- Describe the building blocks of a data center
- Describe compute system, its components, and its types
- Describe compute virtualization, application virtualization, and desktop virtualization
- Provide an overview of storage and connectivity in a data center
- Provide an overview of software-defined data center

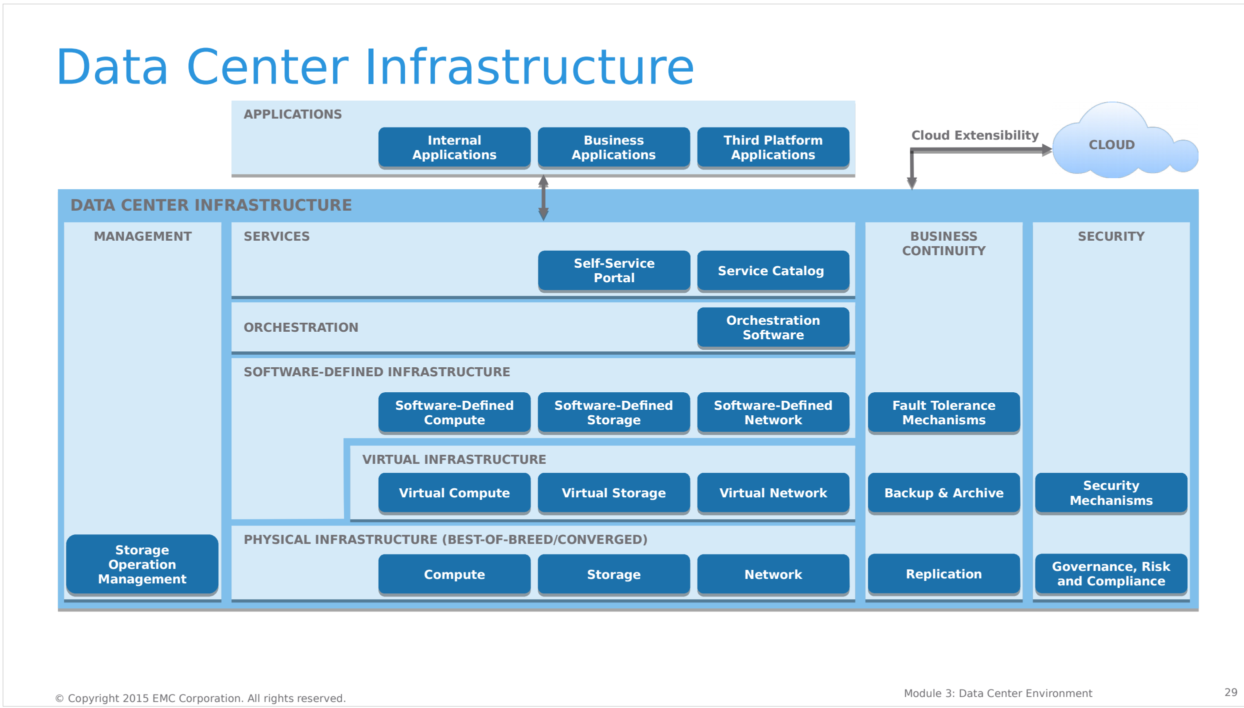
This module focuses on the building blocks of a data center environment. This module also focuses on compute system, its components, and its types. Additionally, this module focuses on compute virtualization, application virtualization, and desktop virtualization. Further, this module focuses on an overview of storage and connectivity in a data center. Finally, this module focuses on an overview of software-defined data center.

Lesson 1: Data Center Infrastructure Building Blocks

This lesson covers the following topics:

- Layers of a data center infrastructure
- Components and functions of each layer
- Cross-layer functions in a data center
- Best-of-breed vs. converged infrastructure

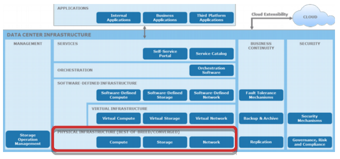
This lesson covers the building blocks of a data center infrastructure. It covers the components and functions of the five layers of a data center. It also covers the three cross-layer functions in a data center. Further, this lesson covers best-of-breed versus converged infrastructure.



Module 1, 'Introduction to Information Storage', defined data center and specified the IT infrastructure and support infrastructure that comprise a data center. The figure on the slide is a block diagram depicting the core IT infrastructure building blocks that make up a data center. The IT infrastructure is arranged in five logical layers and three cross-layer functions. The five layers are physical infrastructure, virtual infrastructure, software-defined infrastructure, orchestration, and services. Each of these layers has various types of hardware and/or software components as shown in the figure. The three cross-layer functions are business continuity, security, and management. Business continuity and security functions include mechanisms and processes that are required to provide reliable and secure access to applications, information, and services. The management function includes various processes that enable

Data Center Infrastructure

Physical Infrastructure

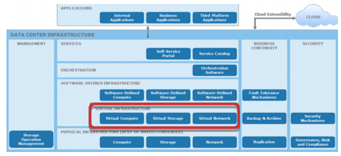


- Foundation layer of the data center infrastructure
- Physical components:
  - Compute systems, storage, and network devices
    - Require operating systems, system software, and protocols for their functions
- Executes the requests generated by the virtual and software-defined layers

The physical infrastructure forms the foundation layer of a data center. It includes equipment such as compute systems, storage systems, and networking devices along with the operating systems, system software, protocols, and tools that enable the physical equipment to perform their functions. A key function of physical infrastructure is to execute the requests generated by the virtual and software-defined infrastructure, such as storing data on the storage devices, performing compute-to-compute communication, executing programs on compute systems, and creating backup copies of data. Compute systems are covered later in this module. Different storage systems are covered in Modules 4, ‘Intelligent Storage Systems (ISS)’, 5, ‘Block-based Storage System’, 6, ‘File-based Storage System (NAS)’, and 7, ‘Object-based and Unified Storage’. Networking is covered in Modules 9, ‘Fibre

Data Center Infrastructure

Virtual Infrastructure



- Virtualization abstracts physical resources and creates virtual resources
- Virtual components:
  - Virtual compute, virtual storage, and virtual network
    - Created from physical resource pools using virtualization software
- Benefits of virtualization:
  - Resource consolidation and multitenant environment
  - Improved resource utilization and increased ROI
  - Flexible resource provisioning and rapid elasticity

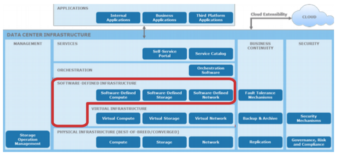
Virtualization is the process of abstracting physical resources, such as compute, storage, and network, and creating virtual resources from them. Virtualization is achieved through the use of virtualization software that is deployed on compute systems, storage systems, and network devices. Virtualization software aggregates physical resources into resource pools from which it creates virtual resources. A resource pool is an aggregation of computing resources, such as processing power, memory, storage, and network bandwidth. For example, storage virtualization software pools the capacity of multiple storage devices to create a single large storage capacity. Similarly, compute virtualization software pools the processing power and memory capacity of a physical compute system to create an aggregation of the power of all processors (in megahertz) and all memory (in megabytes). Examples of virtual resources include

Compute virtualization is covered later in this module, while different storage virtualization and network virtualization techniques are covered later in the course in the storage modules and network modules respectively.

Note: While deploying a data center, an organization may choose not to deploy virtualization. In such an environment, the software-defined layer is deployed directly over the physical infrastructure. Further, it is also possible that part of the infrastructure is virtualized and rest is not virtualized.

Data Center Infrastructure

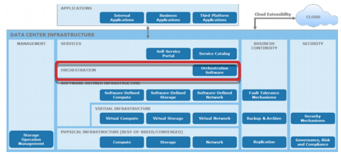
Software-defined Infrastructure



- Deployed either on virtual layer or on physical layer
- All infrastructure components are virtualized and aggregated into pools
  - Underlying resources are abstracted from applications
  - Enables ITaaS
- Centralized, automated, and policy-driven management and delivery of heterogeneous resources
- Components:
  - Software-defined compute
  - Software-defined storage
  - Software-defined network

The software-defined infrastructure layer is deployed either on the virtual layer or on the physical layer. In the software-defined approach, all infrastructure components are virtualized and aggregated into pools. This abstracts all underlying resources from applications. The software-defined approach enables ITaaS, in which consumers provision all infrastructure components as services. It centralizes and automates the management and delivery of heterogeneous resources based on policies. The key architectural components in the software-defined approach include software-defined compute (equivalent to compute virtualization), software-defined storage (SDS), and software-defined network (SDN). Software-defined data center is covered later in this module. Software-defined storage is covered in Module 8, whereas software-defined network is covered in the network modules.

Data Center Infrastructure



Orchestration

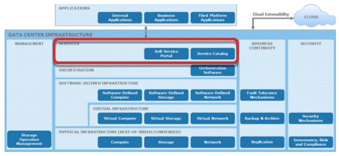
- Component:
  - Orchestration software
- Provides workflows for executing automated tasks
- Interacts with various components across layers and functions to invoke provisioning tasks

The orchestration layer includes the orchestration software. The key function of this layer is to provide workflows for executing automated tasks to accomplish a desired outcome. Workflow refers to a series of inter-related tasks that perform a business operation. The orchestration software enables this automated arrangement, coordination, and management of the tasks. This helps to group and sequence tasks with dependencies among them into a single, automated workflow.

Associated with each service listed in the service catalog, there is an orchestration workflow defined. When a service is selected from the service catalog, an associated workflow in the orchestration layer is triggered. Based on this workflow, the orchestration software interacts with the components across the software-defined layer and the BC, security, and management functions to invoke the provisioning tasks to be executed by the entities.

Data Center Infrastructure

Services

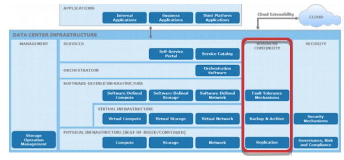


- Delivers IT resources as services to users
  - Enables users to achieve desired business results
  - Users have no liabilities associated with owning the resources
- Components:
  - Service catalog
  - Self-service portal
- Functions of service layer:
  - Stores service information in service catalog and presents them to the users
  - Enables users to access services via a self-service portal

Similar to a cloud service, an IT *service* is a means of delivering IT resources to the end users to enable them to achieve the desired business results and outcomes without having any liabilities such as risks and costs associated with owning the resources. Examples of services are application hosting, storage capacity, file services, and email. The service layer is accessible to applications and end users. This layer includes a service catalog that presents the information about all the IT resources being offered as services. The service catalog is a database of information about the services and includes a variety of information about the services, including the description of the services, the types of services, cost, supported SLAs, and security mechanisms. The provisioning and management requests are passed on to the orchestration layer, where the orchestration workflows—to fulfill the requests—are defined

Data Center Infrastructure

Business Continuity



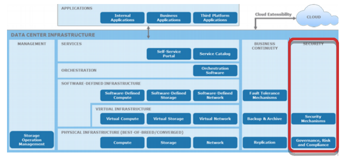
- Enables ensuring the availability of services in line with SLA
- Supports all the layers to provide uninterrupted services
- Includes adoption of measures to mitigate the impact of downtime

Measure	Description
Proactive	• Business impact analysis • Risk assessment • Technology solutions deployment (backup and replication)
Reactive	• Disaster recovery • Disaster restart

The business continuity (BC) cross-layer function specifies the adoption of proactive and reactive measures that enable an organization to mitigate the impact of downtime due to planned and unplanned outages. The proactive measures include activities and processes such as business impact analysis, risk assessment, and technology solutions such as backup, archiving, and replication. The reactive measures include activities and processes such as disaster recovery and disaster restart to be invoked in the event of a service failure. This function supports all the layers—physical, virtual, software-defined, orchestration, and services—to provide uninterrupted services to the consumers. The BC cross-layer function of a cloud infrastructure enables a business to ensure the availability of services in line with the service level agreement (SLA). BC and BC solutions are covered in Modules 12, 'Introduction to Business

Data Center Infrastructure

Security



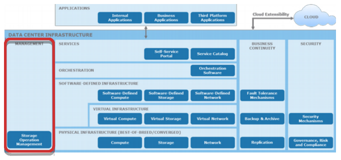
- Supports all the layers to provide secure services
- Specifies the adoption of:
  - Administrative mechanisms
    - Security and personnel policies
    - Standard procedures to direct safe execution of operations
  - Technical mechanisms
    - Firewall
    - Intrusion detection and prevention systems
    - Antivirus
- Security mechanisms enable to meet governance, risk, and compliance (GRC) requirements

The security cross-layer function supports all the infrastructure layers—physical, virtual, software-defined, orchestration, and service—to provide secure services to the consumers. Security specifies the adoption of administrative and technical mechanisms that mitigate or minimize the security threats and provide a secure data center environment. Administrative mechanisms include security and personnel policies or standard procedures to direct the safe execution of various operations. Technical mechanisms are usually implemented through tools or devices deployed on the IT infrastructure. Examples of technical mechanisms include firewall, intrusion detection and prevention systems, and antivirus software.

Governance, risk, and compliance (GRC) specifies processes that help an organization in ensuring that their acts are ethically correct and in accordance with their risk appetite (the risk level

Data Center Infrastructure

Management



- Enables storage infrastructure configuration and capacity provisioning
- Enables problem resolution
- Enables capacity and availability management
- Enables compliance conformance
- Enables monitoring services

The management cross-layer function specifies the adoption of activities related to data center operations management. Adoption of these activities enables an organization to align the creation and delivery of IT services to meet their business objectives. This course focuses on the aspect of storage infrastructure management.

Storage operation management enables IT administrators to manage the data center infrastructure and services. Storage operation management tasks include handling of infrastructure configuration, resource provisioning, problem resolution, capacity, availability, and compliance conformance. This function supports all the layers to perform monitoring, management, and reporting for the entities of the infrastructure. Storage infrastructure management is covered in Module 16, ‘Managing the Storage Infrastructure’.

Best-of-breed Vs. Converged Infrastructure

Best-of-breed infrastructure	Converged infrastructure
• Integrating different best-of-breed components from multiple vendors • Prevents vendor lock-in • Enables repurposing the existing infrastructure components	• Integrates all hardware and software components into a single package • Offers a preconfigured and optimized self-contained unit • Facilitates faster acquisition and deployment

There are two options for building the data center infrastructure – by integrating best-of-breed infrastructure components, or by acquiring and deploying a converged infrastructure.

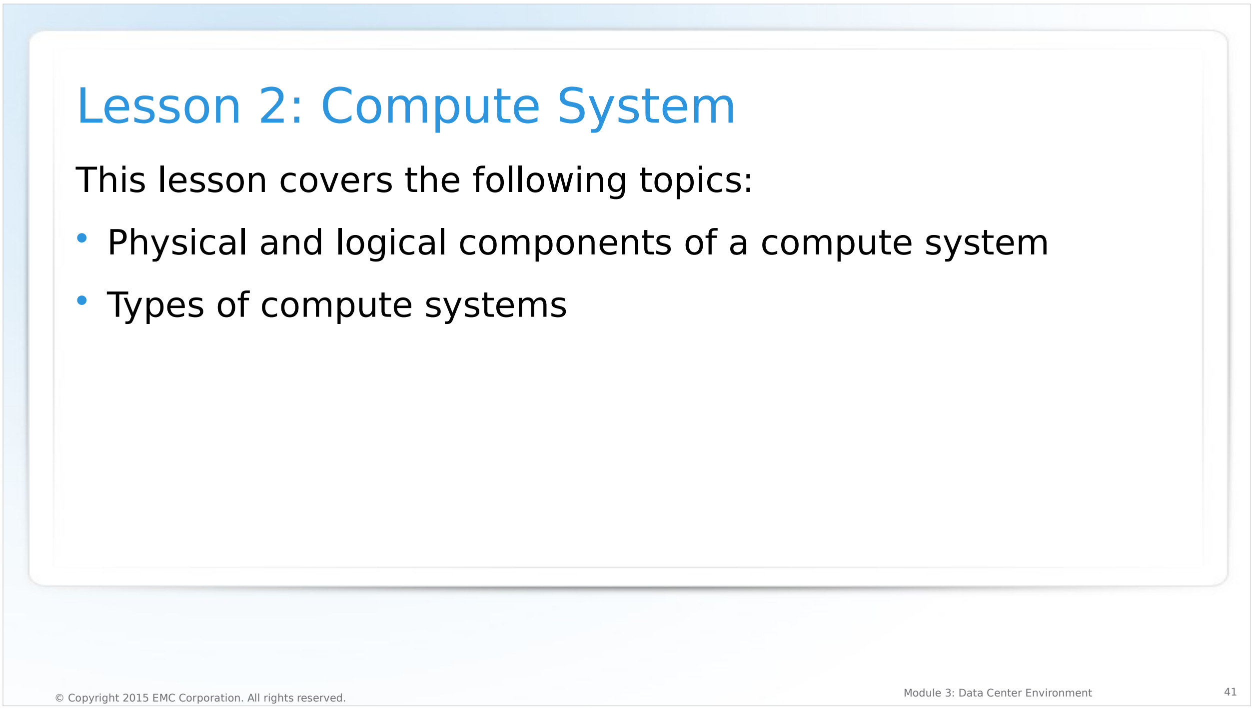
Best-of-breed infrastructure: In this approach, organizations integrate the best-of-breed infrastructure components (hardware and software) purchased from multiple different vendors. This enables the organizations to leverage the advantages of high quality products and services from the respective leading vendors in the segment. It provides the flexibility to change the individual vendors in case the committed support is not provided and the SLAs are not met. Additionally, this approach allows organizations to repurpose the existing infrastructure components, providing a cost benefit. However, this approach requires significant CAPEX, OPEX, and time as it involves evaluation, purchase, testing, deployment, configuration, and integration of

Lesson 1: Summary

During this lesson the following topics were covered:

- Layers of a data center infrastructure
- Components and functions of each layer
- Cross-layer functions of a data center
- Best-of-breed vs. converged infrastructure

This lesson covered the building blocks of a data center infrastructure. It covered the components and functions of the five layers of a data center. It also covered the three cross-layer functions of a data center. Further, this lesson covered best-of-breed versus converged infrastructure.



This lesson covers compute system, and its key physical and logical components. This lesson also covers the types of compute systems.

What is a Compute System?

- A computing platform (hardware and system software) that runs applications
  - Physical components include processor, memory, internal storage, and I/O devices
  - Logical components include OS, device drivers, file system, and logical volume manager

A compute system is a computing device (combination of hardware, firmware, and system software) that runs business applications. Examples of compute systems include physical servers, desktops, laptops, and mobile devices. As mentioned previously in Module 1, 'Introduction to Information Storage' in this course, the term compute system refers to physical servers and hosts on which platform software, management software, and business applications of an organization are deployed.

A compute system's hardware consists of processor(s), memory, internal storage, and I/O devices. The logical components of a compute system include the operating system (OS), file system, logical volume manager, and device drivers. The OS may include the other software or they can be installed individually.

In an enterprise data center, applications are typically deployed on compute clusters for high availability and for balancing computing workloads. A compute cluster is a group of two or more compute systems that function together, sharing certain network and storage resources, and logically viewed as a single system. Compute clustering is covered in detail in Module 12, 'Introduction to Business Continuity'.

Physical Components of a Compute System

Processor	An IC that executes software programs by performing arithmetical, logical, and input/output operations
Random-Access Memory	Volatile data storage that contains the programs for execution and the data used by the processor
Read-Only Memory	Semiconductor memory containing boot, power management, and other device-specific firmware
Motherboard	A PCB that holds the processor, RAM, ROM, network and I/O ports, and other integrated components, such as GPU and NIC
Chipset	A collection of microchips on a motherboard to manage specific functions, such as processor access to RAM and to peripheral ports
Secondary Storage	A persistent storage device such as HDD or SSD

A compute system comprises multiple physical hardware components assembled inside a metal enclosure. Some key components are described below.

Processor: A processor, also known as a Central Processing Unit (CPU), is an integrated circuit (IC) that executes the instructions of a software program by performing fundamental arithmetical, logical, and input/output operations. A common processor/instruction set architecture is the x86 architecture with 32-bit and 64-bit processing capabilities. Modern processors have multiple cores (independent processing units), each capable of functioning as an individual processor.

Random-Access Memory (RAM): The RAM or main memory is an IC that serves as a volatile data storage internal to a compute system. The RAM is directly accessible by the processor, and holds the software programs for the execution and the data used by the processor.

Read-Only Memory (ROM): A ROM is a type of non-volatile semiconductor memory from which data can only be read but not written to. It contains the boot firmware (that enables a compute system to start), power management firmware, and other device-specific firmware.

Motherboard: A motherboard is a printed circuit board (PCB) to which all compute system components connect. It has sockets to hold components such as the microprocessor chip, RAM, and ROM. It also has network ports, I/O ports to connect devices such as keyboard, mouse, and printers, and essential circuitry to carry out computing operations. A motherboard may additionally have integrated components, such as a graphics processing unit (GPU), a network interface card (NIC), and adapters to connect to external storage devices. Motherboards (and other internal components) receive power from a power supply unit.

(Cont'd)

Chipset: A chipset is a collection of microchips on a motherboard and it is designed to perform specific functions. The two key chipset types are Northbridge and Southbridge. Northbridge manages processor access to the RAM and the GPU, while Southbridge connects the processor to different peripheral ports, such as USB ports.

Secondary storage: Secondary storage is a persistent storage device, such as a hard disk drive or a solid state drive, on which the OS and the application software are installed. The processor cannot directly access secondary storage. The desired applications and data are loaded from the secondary storage on to the RAM to enable the processor to access them.

Based on business and performance requirements, cost, and expected rate of growth, an organization has to make multiple important decisions about the choice of compute system hardware to be deployed in a data center. These decisions include the number of compute systems to deploy, the number, the type, and the speed of processors, the amount of RAM required, the motherboard's RAM capacity, the number and type of expansion slots on a motherboard, the number and type of I/O cards, and installation and configuration effort.

Logical Components of a Compute System

- Operating system
- Virtual memory
- Logical volume manager
- File system

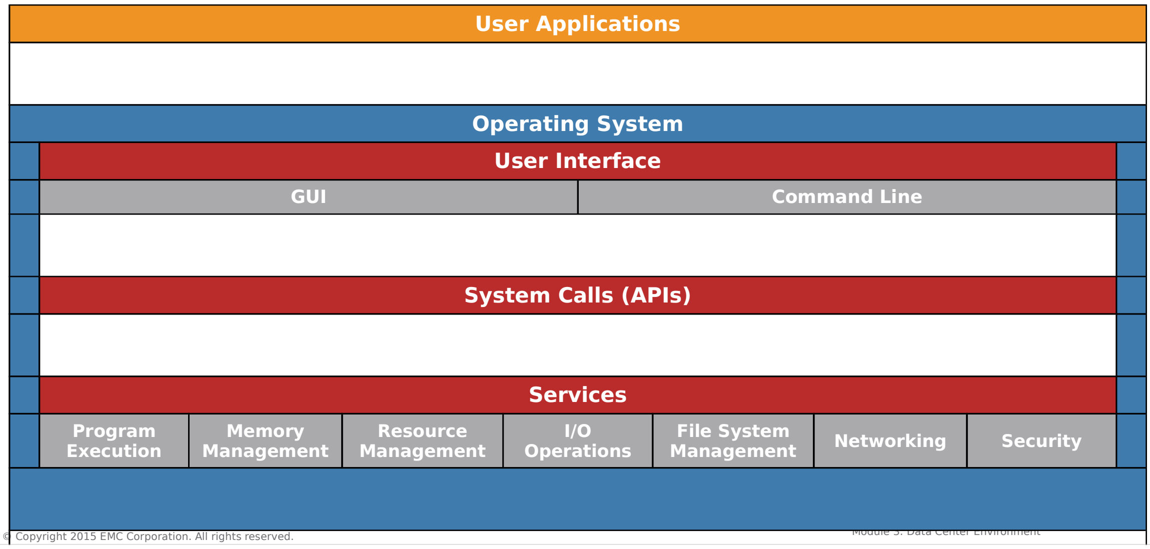
The key logical components of a compute system are:

- Operating system
- Virtual memory
- Logical volume manager
- File system

A detailed description of the logical components is beyond the scope of this course. However, the components are covered in brief next.

Logical Components of a Compute System

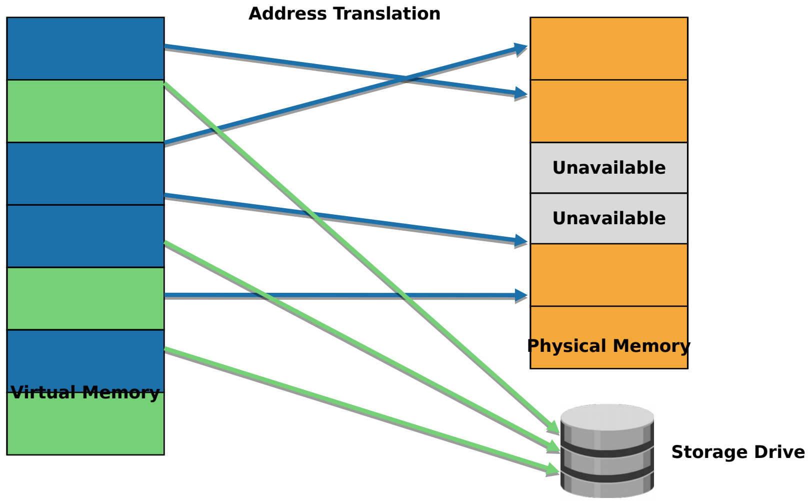
Operating System



The *operating system* (OS) is a software that acts as an intermediary between a user of a compute system and the compute system hardware. It controls and manages the hardware and software on a compute system. The OS manages hardware functions, applications execution, and provides a user interface (UI) for users to operate and use the compute system. The figure on the slide depicts a generic architecture of an OS. Some functions (or services) of an OS include program execution, memory management, resources management and allocation, and input/output management. An OS also provides networking and basic security for the access and usage of all managed resources. It also performs basic storage management tasks while managing other underlying components, such as the device drivers, logical volume manager, and file system. An OS also contains high-level Application Programming Interfaces

Logical Components of a Compute System

Virtual Memory



The amount of physical memory (RAM) in a compute system determines both the size and the number of applications that can run on the compute system. Memory virtualization presents physical memory to applications as a single logical collection of contiguous memory locations called *virtual memory*. While executing applications, the processor generates logical addresses (virtual addresses) that map into the virtual memory. The memory management unit of the processor then maps the virtual address to the physical address. The OS utility, known as the *virtual memory manager* (VMM), manages the virtual memory and also the allocation of physical memory to virtual memory.

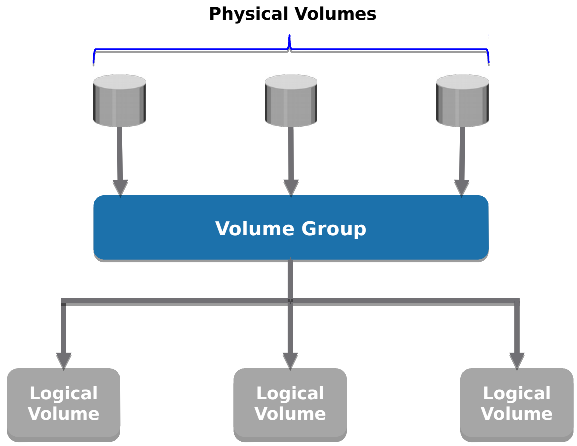
An additional memory virtualization feature of an OS enables the capacity of secondary storage devices to be allocated to the virtual memory. This creates a virtual memory with an address space that is much larger than the actual physical memory space present in the compute system. This enables multiple applications and processes, whose aggregate memory requirement is greater than the available physical memory, to run on a compute system without impacting each other. The VMM manages the virtual-to-physical memory mapping and fetches data from the secondary storage when a process references a virtual address that points to data at the secondary storage. The space used by the VMM on the secondary storage is known as a swap space. A *swap space* (also known as *page file* or *swap file*) is a portion of the storage drive that is used as physical memory.

In a virtual memory implementation, the memory of a system is divided into contiguous blocks of fixed-size pages. A process known as *paging* moves inactive physical memory pages onto the swap file and brings them back to the physical memory when required. This enables efficient use of the available physical memory among different applications. The OS typically moves the least-used pages into the swap file so that enough RAM is available for processes that are more active. The access to swap file pages is slower than physical memory pages because swap file pages are allocated on the storage drive, which is slower than the physical memory.

Logical Components of a Compute System

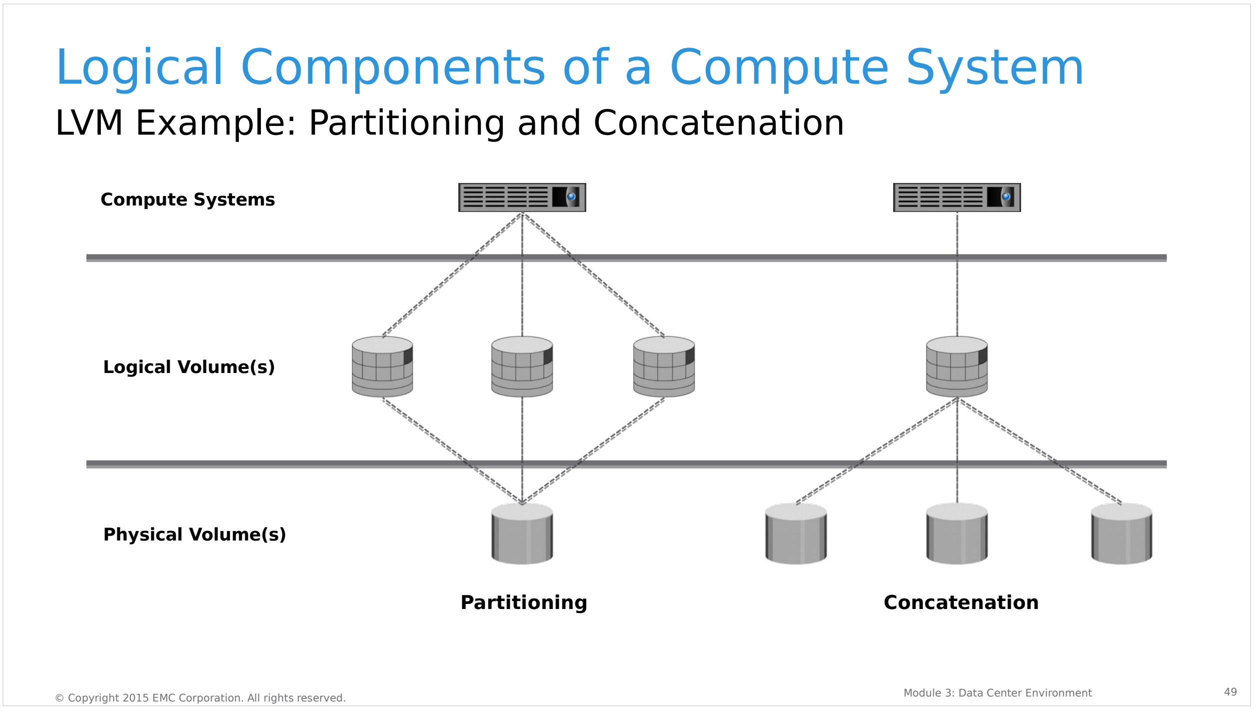
Logical Volume Manager (LVM)

- Creates and controls compute level logical storage
  - Provides a logical view of physical storage
  - Logical data blocks are mapped to physical data blocks
- Physical volumes form a volume group
  - LVM manages volume groups as a single entity
- Logical volumes are created from a volume group



Logical Volume Manager (LVM) is software that runs on a compute system and manages logical and physical storage. LVM is an intermediate layer between the file system and the physical drives. It can partition a larger-capacity disk into virtual, smaller-capacity volumes (*partitioning*) or aggregate several smaller disks to form a larger virtual volume (*concatenation*). LVMs are mostly offered as part of the OS. Earlier, an entire storage drive would be allocated to the file system or the other data entity used by the OS or application. The disadvantage of this was a lack of flexibility. When a storage drive ran out of space, there was no easy way to extend the file system's size. As the storage capacity of the disk drive increased, allocating the entire disk drive for the file system often resulted in underutilization of the storage capacity. The evolution of LVMs enabled dynamic extension of file system capacity and efficient storage management. The LVM provides optimized storage access and simplifies storage resource management. It hides details about the physical disk and the location of data on the disk. It enables administrators to change the storage allocation even when the application is running.

The basic LVM components are physical volumes, logical volume groups, and logical volumes. In LVM terminology, each physical disk connected to the compute system is a *physical volume* (PV). A *volume group* is created by grouping together one or more PVs. A unique *physical volume identifier* (PVID) is assigned to each PV when it is initialized for use by the LVM. Physical volumes can be added or removed from a volume group dynamically. They cannot be shared between different volume groups; which means, the entire PV becomes part of a volume group. Each PV is divided into equal-sized data blocks called *physical extents* when the volume group is created. *Logical volumes* (LV) are created within a given volume group. A LV can be thought of as a disk partition, whereas the volume group itself can be thought of as a disk. The size of a LV is based on a multiple of the number of physical extents. The LV appears as a physical device to the OS. A LV is made up of noncontiguous physical extents and may span over multiple physical volumes. A file system is created on a logical volume. These LVs are then assigned to the application. A logical volume can also be mirrored to provide enhanced data availability.



Disk partitioning was introduced to improve the flexibility and utilization of disk drives. In *partitioning*, a disk drive is divided into logical containers called logical volumes. For example, a large physical drive can be partitioned into multiple LVs to maintain data according to the file system and application requirements. The partitions are created from groups of contiguous cylinders when the hard disk is initially set up on the host. The host's file system accesses the logical volumes without any knowledge of partitioning and physical structure of the disk. *Concatenation* is the process of grouping several physical drives and presenting them to the host as one big logical volume.

Logical Components of a Compute System

File System

- File is a collection of related records stored as a single named unit in contiguous logical address space
- A file system controls and manages the storage and retrieval of files
  - Enables users to perform various operations on files
  - Groups and organizes files in a hierarchical structure
- File system may be broadly classified as:
  - Disk-based file system
  - Network-based file system
  - Virtual file system

A *file* is a collection of related records or data stored as a single named unit in contiguous logical address space. Files are of different types, such as text, executable, image, audio/video, binary, library, and archive. Files have a number of attributes, such as name, unique identifier, type, size, location, owner, and protection.

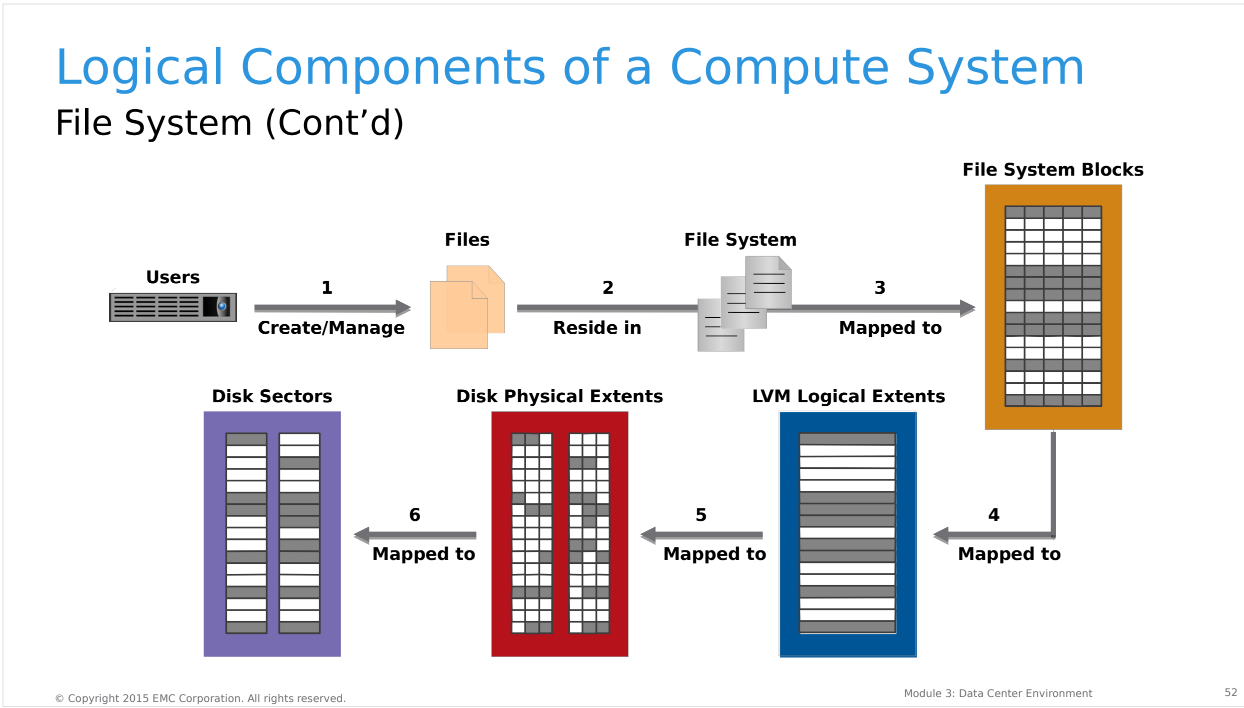
A *file system* is an OS component that controls and manages the storage and retrieval of files in a compute system. A file system enables easy access to the files residing on a storage drive, a partition, or a logical volume. It consists of logical structures and software routines that control access to files. It enables users to perform various operations on files, such as create, access (sequential/random), write, search, edit, and delete.

A file system typically groups and organizes files in a tree-like hierarchical structure. It enables users to

File system may be broadly classified as follows disk-based, network-based, and virtual file systems. These are described below.

Disk-based file system: A disk-based file system manages the files stored on storage devices such as solid-state drives, disk drives, and optical drives. Examples of disk-based file systems are Microsoft NT File System (NTFS), Apple Hierarchical File System (HFS) Plus, Extended File System family for Linux, Oracle ZFS, and Universal Disk Format (UDF).

Network-based file system: A network-based file system uses networking to allow file system access between compute systems. Network-based file systems may use either the client-server model, or may be distributed/clustered. In the client-server model, the file system resides on a server, and is accessed by clients over the network. The client-server model allows clients to mount the remote file systems from the server. NFS for UNIX environment and CIFS for Windows environment (both covered in Module 6, 'File-based Storage System (NAS)') are two standard client-server file sharing protocols. A *clustered file system* is a file system that is simultaneously mounted on multiple compute systems (or nodes) in a cluster. It allows the nodes in the cluster to share and concurrently access the same storage device. Clustered file systems provide features like location-independent addressing and redundancy.



The following is the process of mapping user files to the storage that uses an LVM:

1. Files are created and managed by users and applications.
2. These files reside in the file systems.
3. The file systems are mapped to file system blocks.
4. The file system blocks are mapped to logical extents of a logical volume.
5. These logical extents in turn are mapped to the physical extents either by the OS or by the LVM.
6. These physical extents are mapped to the sectors in a storage subsystem.

If there is no LVM, then there are no logical extents. Without LVM, file system blocks are directly mapped to sectors.

Apart from the files and directories, the file system also includes a number of other related records, which are collectively called the *metadata*. The metadata of a file system must be consistent for the

Types of Compute Systems

Tower Compute System	Rack-mounted Compute System	Blade Compute System
		

The compute systems used in building data centers are typically classified into three categories: tower compute system, rack-mounted compute system, and blade compute system

A tower compute system, also known as a tower server, is a compute system built in an upright standalone enclosure called a “tower”, which looks similar to a desktop cabinet. Tower servers have a robust build, and have integrated power supply and cooling. They typically have individual monitors, keyboards, and mice. Tower servers occupy significant floor space and require complex cabling when deployed in a data center. They are also bulky and a group of tower servers generate considerable noise from their cooling units. Tower servers are typically used in smaller environments. Deploying a large number of tower servers in large environments may involve substantial expenditure.

A rack-mounted compute system, also known as a *rack server*, is a compute system designed to be fixed inside a frame called a “rack”. A *rack* is a standardized enclosure containing multiple mounting slots called “bays”, each of which holds a server in place with the help of screws. A single rack contains multiple servers stacked vertically in bays, thereby simplifying network cabling, consolidating network equipment, and reducing the floor space use. Each rack server has its own power supply and cooling unit. Typically, a console is mounted on a rack to enable administrators to manage all the servers in the rack. Some concerns with rack servers are that they are cumbersome to work with, and they generate a lot of heat because of which more cooling is required, which in turn increases power costs. A “rack unit” (denoted by U or RU) is a unit of measure of the height of a server designed to be mounted on a rack. One rack unit is 1.75 inches (44.45 mm). A 1 U rack server is typically 19 inches (482.6 mm) wide. The standard rack cabinets are 19 inches wide and the common rack cabinet sizes are 42U, 37U, and 27U. The rack cabinets are also used to house network, storage, telecommunication, and other equipment modules. A rack cabinet may also contain a combination of different types of equipment modules.

(Cont’d)

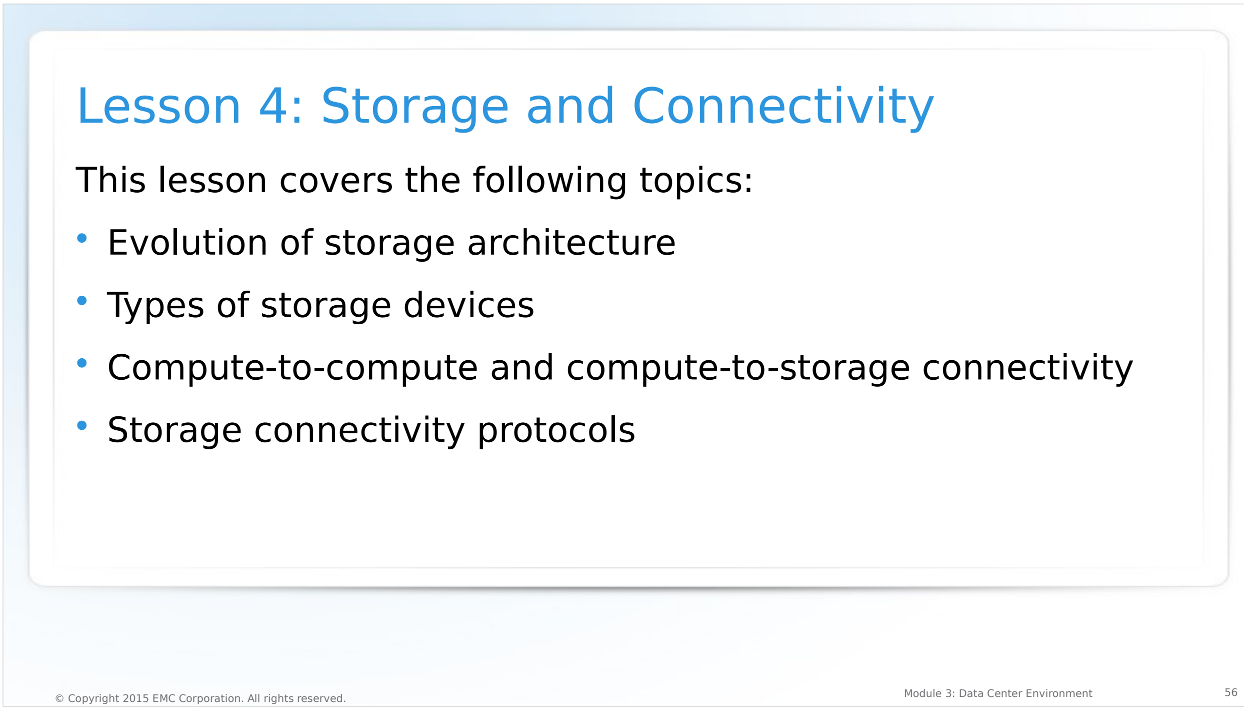
A blade compute system, also known as a *blade server*, is an electronic circuit board containing only core processing components, such as processor(s), memory, integrated network controllers, storage drive, and essential I/O cards and ports. Each blade server is a self-contained compute system and is typically dedicated to a single application. A blade server is housed in a slot inside a blade enclosure (or chassis), which holds multiple blades and provides integrated power supply, cooling, networking, and management functions. The blade enclosure enables interconnection of the blades through a high-speed bus and also provides connectivity to external storage systems. The modular design of the blade servers makes them smaller, which minimizes the floor space requirements, increases the compute system density and scalability, and provides better energy efficiency as compared to the tower and the rack servers. It also reduces the complexity of the compute infrastructure and simplifies compute infrastructure management. It provides these benefits without compromising on any capability that a non-blade compute system provides. Some concerns with blade servers include the high cost of a blade system (blade servers and chassis), and the proprietary architecture of most blade systems due to which a blade server can typically be plugged only into a chassis from the same vendor.

Lesson 2: Summary

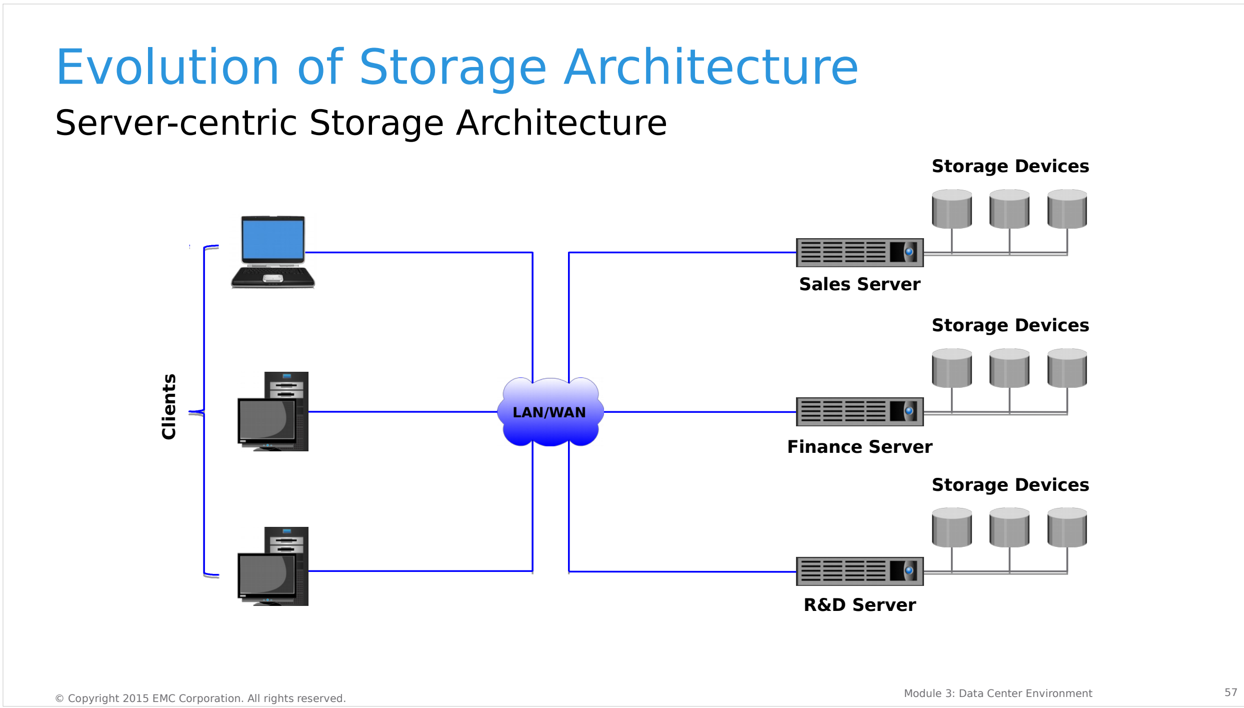
During this lesson the following topics were covered:

- Physical and logical components of a compute system
- Types of compute systems

This lesson covered compute system, and its key physical and logical components. This lesson also covered the types of compute systems.

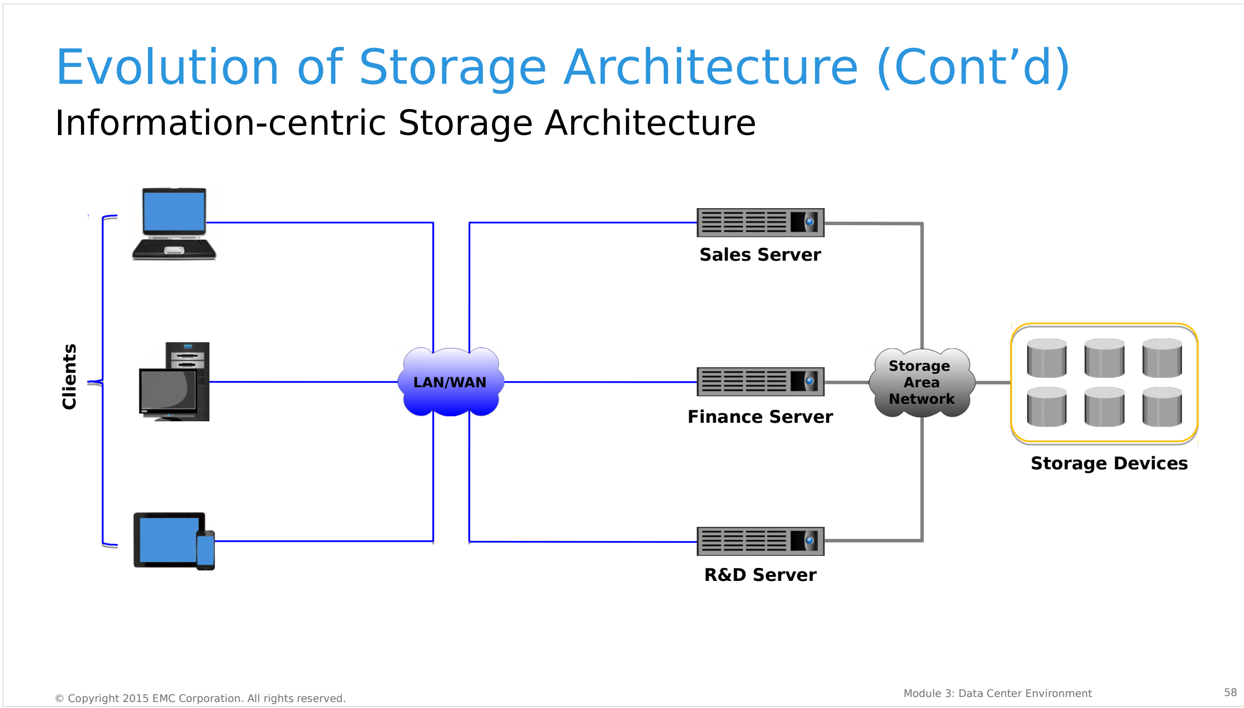


This lesson covers evolution of storage architecture and the types of storage devices. This lesson also covers compute-to-compute and compute-to-storage connectivity. Further, this lesson covers different storage connectivity protocols.

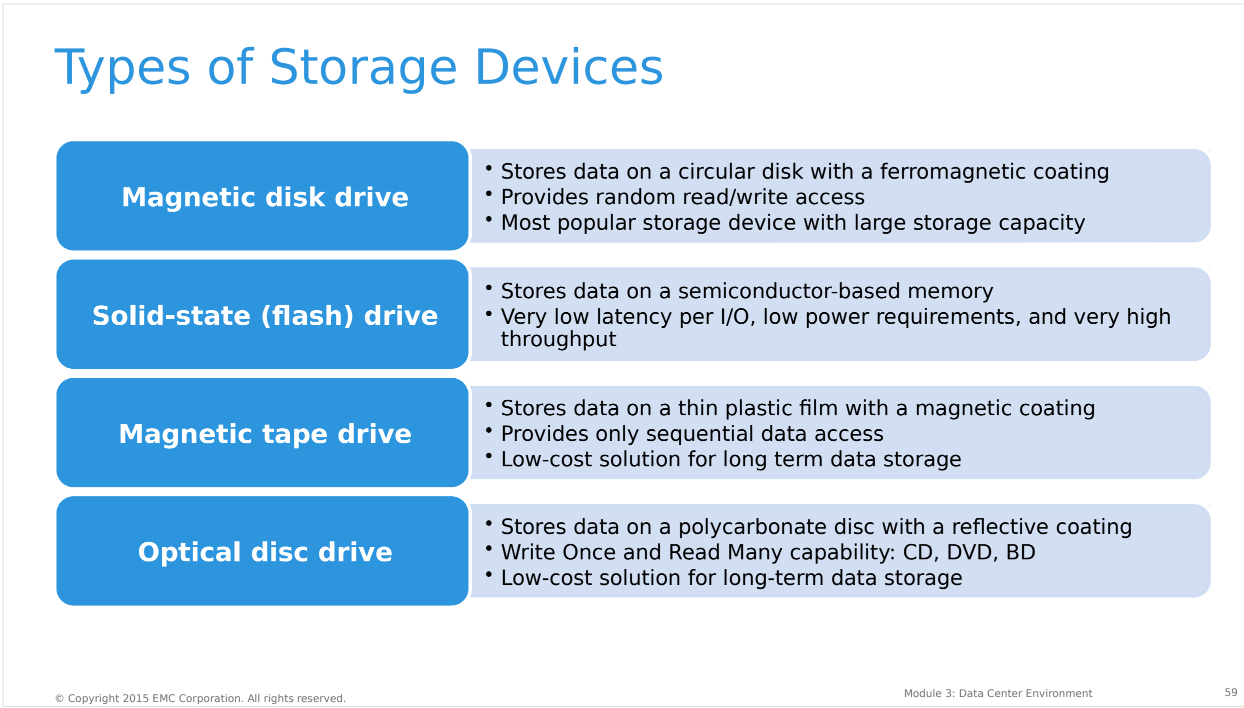


In a traditional environment, business units/departments in an organization have their own servers running the business applications of the respective business unit/department. Storage devices are connected directly to the servers and are typically internal to the server. These storage devices cannot be shared with any other server. This is called *server-centric storage architecture*. In this architecture, each server has a limited number of storage devices, and each storage device exists only in relation to the server to which it is connected. The figure on the slide depicts an example of server-centric architecture. In the figure, the servers of different departments in an organization have directly-connected storage and clients connect to the servers over a local area network (LAN) or a wide area network (WAN).

Traditional server-centric architecture has several



To overcome the challenges of the server-centric architecture, storage evolved to the information-centric architecture. In information-centric architecture, storage devices exist completely independently of servers, and are managed centrally and shared between multiple compute systems. Storage devices assembled within storage systems form a storage pool, and several compute systems access the same storage pool over a specialized, high-speed storage area network (SAN). A SAN is used for information exchange between compute systems and storage systems, and for connecting storage systems. It enables compute systems to share storage resources, improve the utilization of storage systems, and facilitate centralized storage management. SANs are classified based on protocols they support. Common SAN deployment types are Fibre Channel SAN (FC SAN), Internet Protocol SAN (IP SAN), and Fibre Channel over



A *magnetic disk* is a circular storage medium made of non-magnetic material (typically an alloy) and coated with a ferromagnetic material. Data is stored on both surfaces (top and bottom) of a magnetic disk by polarizing a portion of the disk surface. A disk drive is a device that comprises multiple rotating magnetic disks, called platters, stacked vertically inside a metal or plastic casing. Each platter has a rapidly moving arm to read from and write data to the disk. Disk drives are currently the most popular storage medium for storing and accessing data for performance-intensive applications. Disks support rapid access to random data locations and data can be written or retrieved quickly for a number of simultaneous users or applications. Disk drives use pre-defined protocols, such as Advanced Technology Attachment (ATA), Serial ATA (SATA), Small Computer System Interface (SCSI), Serial

A *magnetic tape* is a thin, long strip of plastic film that is coated with a magnetizable material, such as barium ferrite. The tape is packed in plastic cassettes and cartridges. A tape drive is the device to record and retrieve data on a magnetic tape. Tape drives provide linear sequential read/write data access. A tape drive may be standalone or part of a tape library. A tape library contains one or more tape drives and a storage area where a number of tape cartridges are held in slots. Tape is a popular medium for long-term storage due to its relative low cost and portability. Tape drives are typically used by organizations to store large amounts of data, typically for backup, offsite archiving, and disaster recovery. The low access speed due to the sequential access mechanism, the lack of simultaneous access by multiple applications, and the degradation of the tape surface due to the continuous contact with the read/write head are some of the key limitations of tape.

An *optical disc* is a flat, circular storage medium made of polycarbonate with one surface having a special, reflective coating (such as aluminum). An optical disc drive uses a writing laser to record data on the disc in the form of microscopic light and dark dots. A reading laser reads the dots, and generates electrical signals representing the data. The common optical disc types are compact disc (CD), digital versatile disc (DVD), and Blu-ray disc

Overview of Storage Virtualization

- Abstracts physical storage resources to create virtual storage resources:
  - Virtual volumes
  - Virtual disk files
  - Virtual storage systems
- Storage virtualization software can be:
  - Built into the operating environment of a storage system
  - Installed on an independent compute system
  - Built into a hypervisor

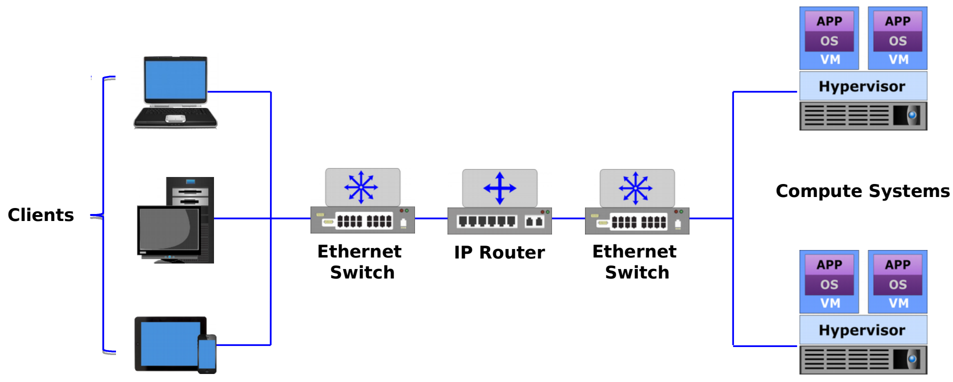
Storage virtualization is the technique of abstracting physical storage resources to create virtual storage resources. Storage virtualization software has the ability to pool and abstract physical storage resources, and present them as a logical storage resources, such as virtual volumes, virtual disk files, and virtual storage systems. Storage virtualization software is either built into the operating environment of a storage system, installed on an independent compute system, or available as hypervisor’s capability. Storage virtualization will be covered in detail in the storage modules of this course.

Introduction to Connectivity

- Communication paths between IT infrastructure components for information exchange and resource sharing
- Types of connectivity:
  - Compute-to-compute connectivity
  - Compute-to-storage connectivity

Connectivity refers to the communication paths between IT infrastructure components for information exchange and resource sharing. The two primary types of connectivity include the interconnection between compute systems, and between a compute system and storage.

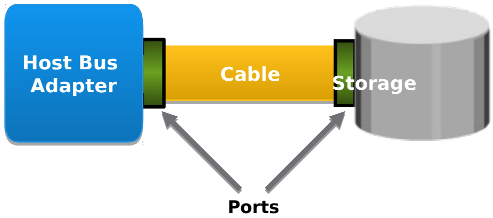
Compute-to-compute Connectivity



Compute-to-compute connectivity typically uses protocols based on the Internet Protocol (IP). Each physical compute system is connected to a network through one or more host interface devices, called a *network interface controller* (NIC). Physical switches and routers are the commonly-used interconnecting devices. A switch enables different compute systems in the network to communicate with each other. A router is an OSI Layer-3 device that enables different networks to communicate with each other. The commonly-used network cables are copper cables and optical fiber cables. The figure on the slide shows a network (LAN or WAN) that provides interconnections among the physical compute systems. It is necessary to ensure that appropriate switches and routers, with adequate bandwidth and ports, are available to provide the required network performance.

Compute-to-storage Connectivity

- Enabled through physical components and interface protocols
- Physical connectivity components:
  - Host bus adapter, port, and cable
- Protocols define formats for communication between devices
  - Popular storage interface protocols are IDE/ATA, SCSI, and FC
- Storage may be connected directly or over a SAN



The discussion in this lesson focuses on the connectivity between compute systems and storage. Storage may be connected directly to a compute system or over a SAN as discussed previously in this lesson. Connectivity and communication between compute and storage are enabled through physical components and interface protocols. The physical components that connect compute to storage are host interface device, port, and cable.

Host bus adapter: A *host bus adapter* (HBA) is a host interface device that connects a compute system to storage or to a SAN. It is an application-specific integrated circuit (ASIC) board that performs I/O interface functions between a compute system and storage, relieving the processor from additional I/O processing workload. A compute system typically contains multiple HBAs.

Storage Connectivity Protocols

Protocol	Description
IDE/ATA	• Popular interface used to connect hard disks and optical drives • The Ultra DMA/133 version of ATA supports a throughput of 133 MB/s
Serial ATA	• Serial version of the IDE/ATA specification typically used for internal connectivity • Provides data transfer rate up to 16 Gb/s (standard 3.2)
SCSI	• Popular standard for compute-to-storage connectivity • Supports up to 16 devices on a single bus • Ultra-640 version provides data transfer speed up to 640 MB/s
SAS	• Point-to-point serial protocol replacing parallel SCSI • Supports data transfer rate up to 12 Gb/s (SAS 3.0)
FC	• Widely-used protocol for high speed compute-to-storage communication • Provides a serial data transmission that operates over copper wire and/or optical fiber • Latest version of the FC interface '16FC' allows transmission of data up to 16 Gb/s
IP	• Existing IP-based network leveraged for storage communication • Examples: iSCSI and FCIP protocols

Integrated Device Electronics (IDE)/Advanced Technology Attachment (ATA) is a popular interface protocol standard used for connecting storage devices, such as disk drives and optical drives. This protocol supports parallel transmission and therefore is also known as Parallel ATA (PATA) or simply ATA. IDE/ATA has a variety of standards and names. The Ultra DMA/133 version of ATA supports a throughput of 133 MB/s. In a master-slave configuration, an ATA interface supports two storage devices per connector. However, if the performance of the drive is important, sharing a port between two devices is not recommended.

The serial version of this protocol supports single bit serial transmission and is known as Serial ATA (SATA). High performance and low cost SATA has largely replaced PATA in the newer systems. SATA revision 3.2 provides a data transfer rate up to 16 Gb/s.

Fibre Channel is a widely-used protocol for high-speed communication to the storage device. The Fibre Channel interface provides gigabit network speed. It provides a serial data transmission that operates over copper wire and optical fiber. The latest version of the FC interface ‘16FC’ allows transmission of data up to 16 Gb/s. The FC protocol and its features are covered in more detail in Module 9, ‘Fibre Channel (FC) SAN’.

IP is a network protocol that has been traditionally used for compute-to-compute traffic. With the emergence of new technologies, an IP network has become a viable option for compute-to-storage communication. IP offers several advantages in terms of cost and maturity and enables organizations to leverage their existing IP-based network. iSCSI and FCIP protocols are common examples that leverage IP for compute-to-storage communication. These protocols are detailed in Module 10, ‘Internet Protocol (IP) SAN’.

Overview of Network Virtualization

- Abstracts physical network resources to create virtual network resources:
  - Virtual switch
  - Virtual LAN
  - Virtual SAN
- Network virtualization software can be:
  - Built into the operating environment of a network device
  - Installed on an independent compute system
  - Built into a hypervisor

Network virtualization is the technique of abstracting physical network resources to create virtual network resources. Network virtualization software is either built into the operating environment of a network device, installed on an independent compute system or available as hypervisor's capability. Network virtualization software has the ability to abstract the physical network resources such as switches and routers to create virtual resources such as virtual switches. It also has the ability to divide a physical network into multiple virtual networks, such as virtual LANs and virtual SANs. Network virtualization available as a hypervisor's capability can emulate the network connectivity between virtual machines (VMs) on a physical compute system. It also enables creating virtual switches that appear to the VMs as physical switches. Network virtualization will be covered later in Module 9, 'Fibre Channel (FC) SAN', 10, 'Internet Protocol (IP) SAN', and 11, 'FC

Lesson 4: Summary

During this lesson the following topics were covered:

- Evolution of storage architecture
- Types of storage devices
- Compute-to-compute and compute-to-storage connectivity
- Storage connectivity protocols

This lesson covered evolution of storage architecture and the types of storage devices. This lesson also covered compute-to-compute and compute-to-storage connectivity. Further, this lesson covered different storage connectivity protocols.