



INFORMATION STORAGE AND MANAGEMENT V2

EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 1

Welcome to Information Storage and Management v2.

Click the Notes tab to view text that corresponds to the audio recording.

Click the Supporting Materials tab to download a PDF version of this eLearning.

Copyright © 1996, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012 EMC Corporation. All Rights Reserved. EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC2, EMC, Data Domain, RSA, EMC Centera, EMC ControlCenter, EMC LifeLine, EMC OnCourse, EMC Proven, EMC Snap, EMC SourceOne, EMC Storage Administrator, Acartus, Access Logix, AdvantEdge, AlphaStor, ApplicationXtender, ArchiveXtender, Atmos, Authentica, Authentic Problems, Automated Resource Manager, AutoStart, AutoSwap, AVALONidm, Avamar, Captiva, Catalog Solution, C-Clip, Celerra, Celerra Replicator, Centera, CenterStage, CentraStar, ClaimPack, ClaimsEditor, CLARiiON, ClientPak, Codebook Correlation Technology, Common Information Model, Configuration Intelligence, Configuresoft, Connectrix, CopyCross, CopyPoint, Dantz, DatabaseXtender, Direct Matrix Architecture, DiskXtender, DiskXtender 2000, Document Sciences, Documentum, elInput, E-Lab, EmailXaminer, EmailXtender, Enginuity, eRoom, Event Explorer, FarPoint, FirstPass, FLARE, FormWare, Geosynchrony, Global File Virtualization, Graphic Visualization, Greenplum, HighRoad, HomeBase, InfoMover, Infoscapes, Infra, InputAccel, InputAccel Express, Invista, Ionix, ISIS, Max Retriever, MediaStor, MirrorView, Navisphere, NetWorker, nLayers, OnAlert, OpenScale, PixTools, Powerlink, PowerPath, PowerSnap, QuickScan, Rainfinity, RepliCare, RepliStor, ResourcePak, Retrospect, RSA, the RSA logo, SafeLine, SAN Advisor, SAN Copy, SAN Manager, Smarts, SnapImage, SnapSure, SnapView, SRDF, StorageScope, SupportMate, SymmAPI, SymmEnabler, Symmetrix, Symmetrix DMX, Symmetrix VMAX, TimeFinder, UltraFlex, UltraPoint, UltraScale, Unisphere, VMAX, Vblock, Viewlets, Virtual Matrix, Virtual Matrix Architecture, Virtual Provisioning, VisualSAN, VisualSRM, Voyence, VPLEX, VSAM-Assist, WebXtender, xPression, xPresso, YottaYotta, the EMC logo, and where information lives, are registered trademarks or trademarks of EMC Corporation in the United States and other countries.

All other trademarks used herein are the property of their respective owners.

© Copyright 2012 EMC Corporation. All rights reserved. Published in the USA.

Revision Date: 06/30/2012

Revision Number:

Course Objectives

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

- Evaluate storage architectures and key data center elements in classic, virtualized, and cloud environments
- Explain physical and logical components of a storage infrastructure including storage subsystems, RAID, and intelligent storage systems
- Describe storage networking technologies such as FC SAN, IP SAN, FCoE, NAS, and object-based and unified storage
- Articulate business continuity solutions – backup and replication, and archive for managing fixed content
- Describe information security requirements and solutions, and identify parameters for managing and monitoring storage infrastructure in classic, virtualized, and cloud environments

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

- Evaluate storage architectures and key data center elements in classic, virtualized, and cloud environments
- Explain physical and logical components of a storage infrastructure including storage subsystems, RAID, and intelligent storage systems
- Describe storage networking technologies such as FC SAN, IP SAN, FCoE, NAS, and object-based and unified storage
- Articulate business continuity solutions—backup and replication, and archive for managing fixed content
- Describe information security requirements and solutions, and identify parameters for managing and monitoring storage infrastructure in classic, virtualized, and cloud environments

Course Organization

- Section 1: Storage System
- Section 2: Storage Networking Technologies
- Section 3: Backup, Archive, and Replication
- Section 4: Cloud Computing
- Section 5: Securing and Managing Storage Infrastructure

This course is divided into five sections. Each section contains one or more modules. All the five sections should be taken in the order to complete the course.

Become a Certified EMC Proven Professional!

Information Storage and Management v2 course supports EMC Proven Professional certification.

The #1 certification program in the information storage and management industry.



EMC Proven Professional certification details

Certification Level	Associate
Exam Name	E10-001 Information Storage and Management Exam Version 2
Credential Name	EMC Proven Professional Information Storage Associate v2 (EMCISA)

Next Steps:

1. **Assess** your level of expertise

Take a Free online practice test to identify areas you may need to review.

2. **Validate** your expertise

Register for your proctored exam at a Pearson VUE testing center near you.

For more information, please visit <http://education.emc.com/certification>.

Why Get Proven?

Being Proven means investing in yourself and formally validating your knowledge, skills, and expertise by the industry's most comprehensive education and certification program.

Get Proven. Join a community of dedicated professionals, share exclusive benefits:

Keep Current – Ensure that your knowledge grows as fast as the pace of technology changes

- Receive no-cost Knowledge Maintenance updates on the latest EMC products and technologies.
- Learn from in-depth technical papers from our Knowledge Sharing library.

Lead and Advocate – Establish yourself as an industry thought-leader and mentor

- Become an advocate for the Information Storage and Management industry.
- Get published. Share your expertise and best practices in our Knowledge Sharing Competition.

Connect and Collaborate – Join a community of the most trusted professionals in the industry

- Discuss, share, find answers, or simply connect in our EMC Proven Professional online community.



Like Us:
[EMC Proven Professional](#)



<http://education.EMC.com/ProvenCommunity>



Follow Us:
[@EMCEducation](#), [@EMCProven](#)



Section 1: Storage System

This section contains four modules:

- Module 1: Introduction to Information Storage
- Module 2: Data Center Environment
- Module 3: Data Protection – RAID
- Module 4: Intelligent Storage System

Welcome to Section 1: Storage System. This section covers four modules, namely Introduction to Information Storage, Data Center Environment, Data Protection – RAID, and Intelligent Storage System.

Module – 1

Introduction to Information Storage



Module 1: Introduction to Information Storage

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

- Define data and information
- Describe types of data
- Describe the evolution of storage architecture
- Describe the core elements of a data center
- List the key characteristics of data center
- Provide an overview of virtualization and cloud computing

This module focuses on the definition of data and information, types of data, and evolution of storage architecture. It lists the five core elements of a data center and describes the key characteristics of a data center. This module also provides an overview of virtualization and cloud computing.

Why Information Storage and Management?

- Information is the knowledge derived from data
- Growth of digital information has resulted in information explosion
- We live in an on-command, on-demand world
 - ▶ We need information when and where required
- Increasing dependency on fast and reliable access to information
- Businesses seek to store, protect, optimize, and leverage the information
 - ▶ To gain competitive advantage
 - ▶ To derive new business opportunity

Information is increasingly important in our daily lives. We have become information-dependent in the 21st century, living in an on-command, on-demand world, which means, we need information when and where it is required. We access the Internet every day to perform searches, participate in social networking, send and receive e-mails, share pictures and videos, and use scores of other applications. Equipped with a growing number of content-generating devices, more information is created by individuals than by organizations (including business, governments, non-profits and so on). Information created by individuals gains value when shared with others. When created, information resides locally on devices, such as cell phones, smartphones, tablets, cameras, and laptops. To be shared, this information needs to be uploaded to central data repository (data centers) via networks. Although the majority of information is created by individuals, it is stored and managed by a relatively small number of organizations.

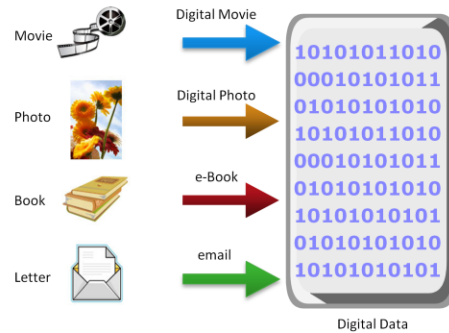
The importance, dependency, and volume of information for the business world also continue to grow at astounding rates. Businesses depend on fast and reliable access to information critical to their success. Examples of business processes or systems that rely on digital information include airline reservations, telecommunications billing, internet commerce, electronic banking, credit card transaction processing, capital/stock trading, health care claims processing, life science research and so on. The increasing dependence of businesses on information has amplified the challenges in storing, protecting, and managing data. Legal, regulatory, and contractual obligations regarding the availability and protection of data further add to these challenges.

What is Data?

Data

It is a collection of raw facts from which conclusions may be drawn.

- Data is converted into more convenient form – digital data
- Factors for digital data growth are:
 - ▶ Increase in data-processing capabilities
 - ▶ Lower cost of digital storage
 - ▶ Affordable and faster communication technology
 - ▶ Proliferation of applications and smart devices



Data is a collection of raw facts from which conclusions may be drawn. Handwritten letters, a printed book, a family photograph, printed and duly signed copies of mortgage papers, a bank's ledgers, and an airline ticket are examples that contain data.

Before the advent of computers, the methods adopted for data creation and sharing were limited to fewer forms, such as paper and film. Today, the same data can be converted into more convenient forms, such as an e-mail message, an e-book, a digital image, or a digital movie. This data can be generated using a computer and stored as strings of binary numbers (0s and 1s). Data in this form is called digital data and is accessible by the user only after a computer processes it.

Businesses analyze raw data to identify meaningful trends. On the basis of these trends, a company can plan or modify its strategy. For example, a retailer identifies customers' preferred products and brand names by analyzing their purchase patterns and maintaining an inventory of those products. Effective data analysis not only extends its benefits to existing businesses, but also creates the potential for new business opportunities by using the information in creative ways.

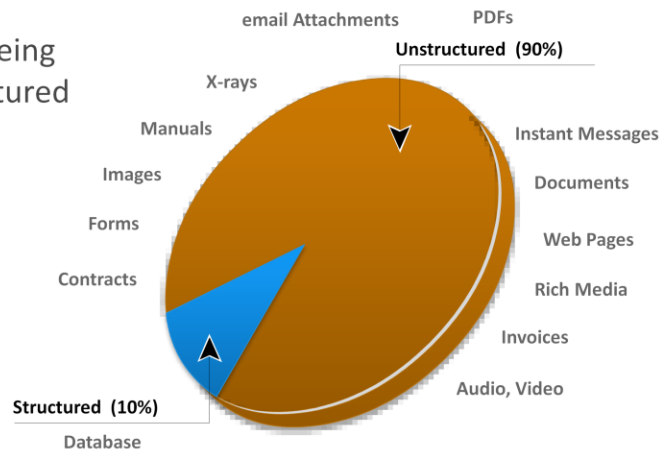
Cont..

With the advancement of computer and communication technologies, the rate of data generation and sharing has increased exponentially. The following is a list of some of the factors that have contributed to the growth of digital data:

- **Increase in data-processing capabilities:** Modern computers provide a significant increase in processing and storage capabilities. This enables the conversion of various types of content and media from conventional forms to digital formats.
- **Lower cost of digital storage:** Technological advances and the decrease in the cost of storage devices have provided low-cost storage solutions. This cost benefit has increased the rate at which digital data is generated and stored.
- **Affordable and faster communication technology:** The rate of sharing digital data is now much faster than traditional approaches. A handwritten letter might take a week to reach its destination, whereas it typically takes only a few seconds for an e-mail message to reach its recipient.
- **Proliferation of applications and smart devices:** Smartphones, tablets, and newer digital devices, along with smart applications, have significantly contributed to the generation of digital content.

Types of Data

- Data can be classified as:
 - ▶ Structured
 - ▶ Unstructured
- Majority of data being created is unstructured



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 12

Data can be classified as structured or unstructured based on how it is stored and managed. Structured data is organized in rows and columns in a rigidly defined format so that applications can retrieve and process it efficiently. Structured data is typically stored using a database management system (DBMS).

Data is unstructured if its elements cannot be stored in rows and columns, which makes it difficult to query and retrieve by applications. For example, customer contacts that are stored in various forms such as sticky notes, e-mail messages, business cards, or even digital format files, such as .doc, .txt, and .pdf. Due to its unstructured nature, it is difficult to retrieve this data using a traditional customer relationship management application. A vast majority of new data being created today is unstructured. The industry is challenged with new architectures, technologies, techniques, and skills to store, manage, analyze, and derive value from unstructured data from numerous sources.

Big Data – 1

Big Data

It refers to data sets whose sizes are beyond the ability of commonly used software tools to capture, store, manage, and process within acceptable time limits.

- Includes both structured and unstructured data generated by variety of sources
- Big data analysis in real time requires new techniques and tools that provide:
 - ▶ High performance
 - ▶ Massively parallel processing (MPP) data platforms
 - ▶ Advanced analytics
- Big data analytics provide an opportunity to translate large volumes of data into right decisions

Big data is a new and evolving concept, which refers to data sets whose sizes are beyond the capability of commonly used software tools to capture, store, manage, and process within acceptable time limits. It includes both structured and unstructured data generated by a variety of sources, including business application transactions, web pages, videos, images, e-mails, social media, and so on. These data sets typically require real-time capture or updates for analysis, predictive modeling, and decision making.

Traditional IT infrastructure and data processing tools and methodologies are inadequate to handle the volume, variety, dynamism, and complexity of big data. Analyzing big data in real time requires new techniques, architectures, and tools that provide high performance, massively parallel processing (MPP) data platforms, and advanced analytics on the data sets.

Big Data – 2

- Data Science is an emerging discipline
 - ▶ Enables organizations to derive business value from big data
 - ▶ Synthesis of several disciplines, such as
 - ▶▶ Statistics, math, data visualization, and computer science

Data Science is an emerging discipline, which enables organizations to derive business value from big data. Data Science represents the synthesis of several existing disciplines, such as statistics, math, data visualization and computer science to enable data scientists to develop advanced algorithms for the purpose of analyzing vast amounts of information to drive new value and make more data-driven decisions. Several industries and markets currently looking to employ data science techniques include medical and scientific research, healthcare, public administration, fraud detection, social media, banks, insurance companies, and other digital information-based entities that benefit from the analytics of big data. The storage architecture required for big data should be simple, efficient, and inexpensive to manage, yet provide access to multiple platforms and data sources simultaneously.

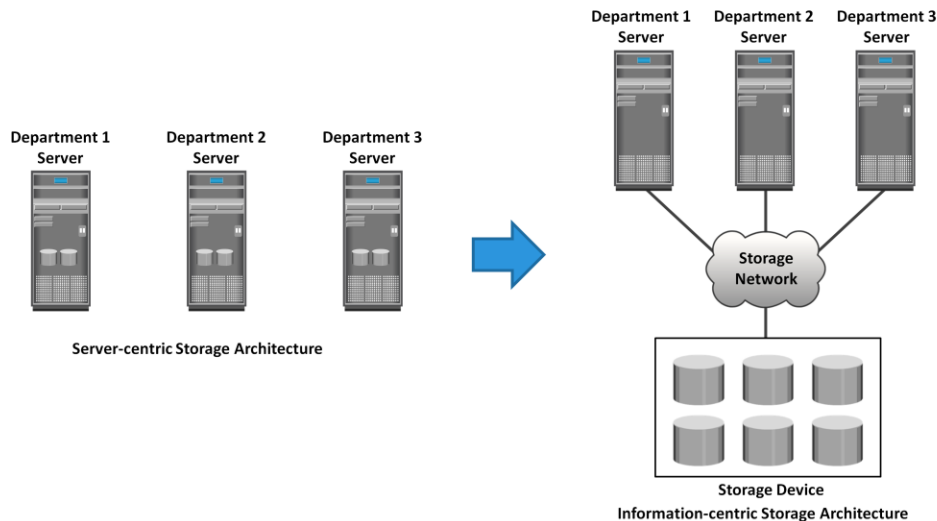
Storage

- Stores data created by individuals and organizations
 - ▶ Provides access to data for further processing
- Examples of storage devices are:
 - ▶ Media card in a cell phone or digital camera
 - ▶ DVDs, CD-ROMs
 - ▶ Disk drives
 - ▶ Disk arrays
 - ▶ Tapes

Data created by individuals or businesses must be stored so that it is easily accessible for further processing. In a computing environment, devices designed for storing data are termed storage devices or simply storage. The type of storage used varies based on the type of data and the rate at which it is created and used. Devices, such as a media card in a cell phone or digital camera, DVDs, CD-ROMs, and disk drives in personal computers are examples of storage devices.

Businesses have several options available for storing data, including internal hard disks, external disk arrays, and tapes.

Evolution of Storage Architecture



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 16

Historically, organizations had centralized computers (mainframes) and information storage devices (tape reels and disk packs) in their data center. The evolution of open systems, their affordability, and ease of deployment made it possible for business units/departments to have their own servers and storage. In earlier implementations of open systems, the storage was typically internal to the server. These storage devices could not be shared with any other servers. This approach is referred to server-centric storage architecture. In this architecture, each server has a limited number of storage devices, and any administrative tasks, such as maintenance of the server or increasing storage capacity, might result in unavailability of information. The proliferation of departmental servers in an enterprise resulted in unprotected, unmanaged, fragmented islands of information and increased capital and operating expenses.

To overcome these challenges, storage evolved from server-centric to information-centric architecture. In this architecture, storage devices are managed centrally and independent of servers. These centrally-managed storage devices are shared with multiple servers. When a new server is deployed in the environment, storage is assigned from the same shared storage devices to that server. The capacity of shared storage can be increased dynamically by adding more storage devices without impacting information availability. In this architecture, information management is easier and cost-effective.

Storage technology and architecture continue to evolve, which enables organizations to consolidate, protect, optimize, and leverage their data to achieve the highest return on information assets.

Data Center

Data Center

It is a facility that contains storage, compute, network, and other IT resources to provide centralized data-processing capabilities.

- Core elements of a data center
 - ▶ Application
 - ▶ Database management system (DBMS)
 - ▶ Host or Compute
 - ▶ Network
 - ▶ Storage
- These core elements work together to address data-processing requirements

Organizations maintain data centers to provide centralized data-processing capabilities across the enterprise. Data centers house and manage large amounts of data. The data center infrastructure includes hardware components, such as computers, storage systems, network devices, and power backups; and software components, such as applications, operating systems, and management software. It also includes environmental controls, such as air conditioning, fire suppression, and ventilation.

Large organizations often maintain more than one data center to distribute data processing workloads and provide backup if a disaster occurs.

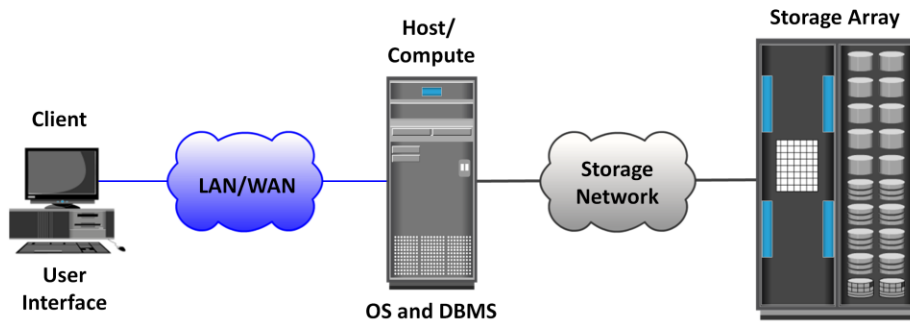
Five core elements are essential for the functionality of a data center:

- **Application:** A computer program that provides the logic for computing operations
- **Database management system (DBMS):** Provides a structured way to store data in logically organized tables that are interrelated
- **Host or compute:** A computing platform (hardware, firmware and software) that runs applications and databases
- **Network:** A data path that facilitates communication among various networked devices
- **Storage:** A device that stores data persistently for subsequent use

These core elements are typically viewed and managed as separate entities, but all the elements must work together to address data-processing requirements.

Note: In this course host, compute, and server are used interchangeably to represent the element that runs applications.

Data Center: Online Order Transaction System Example



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

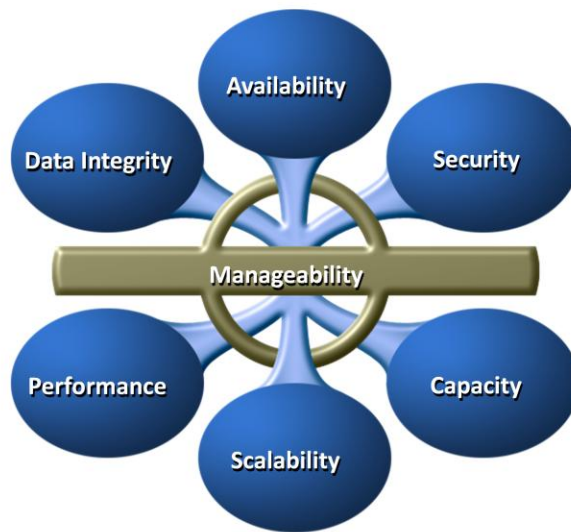
Section 1: Storage System 18

Figure in the slide shows an example of an online order transaction system that involves the five core elements of a data center and illustrates their functionality in a business process.

A customer places an order through a client machine connected over a LAN/WAN to a host running an order-processing application. The client accesses the DBMS on the host through the application to provide order-related information, such as the customer name, address, payment method, products ordered, and quantity ordered.

The DBMS uses the host operating system to write this data to the physical disks in the storage array. The storage networks provide the communication link between the host and the storage array and transports the request to read or write data between them. The storage array, after receiving the read or write request from the host, performs the necessary operations to store the data on physical disks.

Key Characteristics of a Data Center



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 19

Uninterrupted operation of data centers is critical to the survival and success of a business. Although the characteristics shown in the slide are applicable to all elements of the data center infrastructure, the focus here is on storage systems.

- **Availability:** A data center should ensure the availability of information when required. Unavailability of information could cost millions of dollars per hour to businesses, such as financial services, telecommunications, and e-commerce.
- **Security:** Data centers must establish policies, procedures, and core element integration to prevent unauthorized access to information.
- **Scalability:** Business growth often requires deploying more servers, new applications, and additional databases. Data center resources should scale based on requirements, without interrupting business operations.
- **Performance:** All the elements of the data center 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.

Cont..

- **Capacity:** Data center operations require adequate resources to store and process large amounts of data, efficiently. When capacity requirements increase, the data center must provide additional capacity without interrupting availability or with minimal disruption. Capacity may be managed by reallocating the existing resources or by adding new resources.
- **Manageability:** A data center should provide easy and integrated management of all its elements. Manageability can be achieved through automation and reduction of human (manual) intervention in common tasks.

Managing Data Center

- Key management activities include
 - ▶ Monitoring
 - ▶▶ Continuous process of gathering information on various elements and services running in a data center
 - ▶ Reporting
 - ▶▶ Details on resource performance, capacity, and utilization
 - ▶ Provisioning
 - ▶▶ Configuration and allocation of resources to meet the capacity, availability, performance, and security requirements
- Virtualization and cloud computing have changed the way data center infrastructure resources are provisioned and managed

Managing a data center involves many tasks. The key management activities include the following:

- **Monitoring:** It is a continuous process of gathering information on various elements and services running in a data center. The aspects of a data center that are monitored include security, performance, availability, and capacity.
- **Reporting:** It is done periodically on resource performance, capacity, and utilization. Reporting tasks help to establish business justifications and chargeback of costs associated with data center operations.
- **Provisioning:** It is a process of providing the hardware, software, and other resources required to run a data center. Provisioning activities primarily include resources management to meet capacity, availability, performance, and security requirements.

Virtualization and cloud computing have dramatically changed the way data center infrastructure resources are provisioned and managed. Organizations are rapidly deploying virtualization on various elements of data centers to optimize their utilization. Further, continuous cost pressure on IT and on-demand data processing requirements have resulted in the adoption of cloud computing.

Virtualization: An Overview

- Virtualization is a technique of abstracting physical resources and making them appear as logical resources
 - ▶ For example partitioning of raw disks
- Pools physical resources and provides an aggregated view of physical resource capabilities
- Virtual resources can be created from pooled physical resources
 - ▶ Improves utilization of physical IT resources

Virtualization is a technique of abstracting physical resources, such as compute, storage, and network, and making them appear as logical resources. Virtualization existed in the IT industry for several years and in different forms. Common examples of virtualization are virtual memory used on compute systems and partitioning of raw disks.

Virtualization enables pooling of physical resources and providing an aggregated view of the physical resource capabilities. For example, storage virtualization enables multiple pooled storage devices to appear as a single large storage entity. Similarly, by using compute virtualization, the CPU capacity of the pooled physical servers can be viewed as aggregation of the power of all CPUs (in megahertz). Virtualization also enables centralized management of pooled resources.

Virtual resources can be created and provisioned from the pooled physical resources. For example, a virtual disk of a given capacity can be created from a storage pool or a virtual server with specific CPU power and memory can be configured from a compute pool. These virtual resources share pooled physical resources, which improves the utilization of physical IT resources. Based on business requirements, capacity can be added to or removed from the virtual resources without any disruption to applications or users. With improved utilization of IT assets, organizations save the costs associated with procurement and management of new physical resources. Moreover, fewer physical resources means less space and energy, which leads to better economics and green computing.

Cloud Computing: An Overview

- Enables individuals and organizations to use IT resources as a service over network
- Enables self-service requesting and automates request-fulfillment process
 - ▶ Enables users to scale up or scale down the usage of computing resources quickly
- Enables consumption-based metering
 - ▶ Consumers pay only for the resources they use
 - ▶▶ Example: CPU hours used, amount of data transferred, and Gigabytes of data stored

In today's fast-paced and competitive environment, organizations must be agile and flexible to meet changing market requirements. This leads to rapid expansion and upgrade of resources while meeting stagnant IT budgets. Cloud computing addresses these challenges efficiently. Cloud computing enables individuals or businesses to use IT resources as a service over the network. It provides highly scalable and flexible computing that enables provisioning of resources on demand. Users can scale up or scale down the demand of computing resources, including storage capacity, with minimal management effort or service provider interaction. Cloud computing empowers self-service requesting through a fully automated request-fulfillment process. Cloud computing enables consumption-based metering; therefore, consumers pay only for the resources they use, such as CPU hours used, amount of data transferred, and gigabytes of data stored.

Cloud infrastructure is usually built upon virtualized data centers, which provide resource pooling and rapid provisioning of resources. Information storage in virtualized and cloud environments is detailed later in this course.

Module 1: Summary

Key points covered in this module:

- Data and information
- Types of data
- Big data
- Evolution of storage architecture
- Core elements of data center
- Key characteristics of data center
- Virtualization and cloud computing

This module covered the definition of data and information. Data is a collection of raw facts from which conclusions may be drawn and information is the intelligence and knowledge derived from data. Businesses analyze raw data to identify meaningful trends. On the basis of these trends, a company can plan or modify its strategy.

Data can be classified as structured and unstructured. Big data refers to data sets whose sizes are beyond the ability of commonly used software tools to capture, store, manage, and process within acceptable time limits.

Information-centric architecture is commonly deployed in today's data center. It helps to overcome the challenges of server-centric storage architecture.

A data center has five core elements such as application, database management system (DBMS), host, network, and storage.

The key characteristics of data are availability, security, scalability, performance, data integrity, capacity, and manageability.

Virtualization is a technique of abstracting physical resources, such as compute, storage, and network, and making them appear as logical resources.

Cloud computing enables individuals or businesses to use IT resources as a service over the network.

Check Your Knowledge – 1

- Which is an example of structured data?
 - A. Image
 - B. PDF document
 - C. Database
 - D. Web page
- Which is true about big data?
 - A. Includes only unstructured data
 - B. Includes data from a single source
 - C. Captured efficiently using traditional software tools
 - D. Data size is beyond the capability of traditional software to process

Check Your Knowledge – 2

- Which is a feature of information-centric architecture?
 - A. Storage is internal to the servers
 - B. Prevents sharing of storage among servers
 - C. Consists of server, network, and storage in a single system
 - D. Storage is managed centrally and independent of servers

- What accurately describes virtualization?
 - A. Provides on-demand, metered services
 - B. Abstracts physical resources into logical resources
 - C. Pools logical resources to provide data integrity
 - D. Enables decentralized management across data centers

Check Your Knowledge – 3

- Which requirement refers to the ability of a storage solution to grow with the business?
 - A. Availability
 - B. Manageability
 - C. Integrity
 - D. Scalability

Module – 2

Data Center Environment



Module 2: Data Center Environment

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

- Describe the core elements of a data center
- Describe virtualization at application and host layer
- Describe disk drive components and performance
- Describe host access to storage through DAS
- Describe working and benefits of flash drives

This module focuses on the key components of a data center. It also includes virtualization at compute, memory, desktop, and application. Storage and network virtualization are discussed later in the course. This module also focuses on storage subsystems and provides details on components, geometry, and performance parameters of a disk drive. The connectivity between the host and storage facilitated by various technologies is also explained.

Module 2: Data Center Environment

Lesson 1: Application, DBMS, and Host (Compute)

During this lesson the following topics are covered:

- Application and application virtualization
- DBMS
- Components of host system
- Compute and memory virtualization

This lesson covers three key components of a data center – application, DBMS, and compute. Hardware and software components of a compute system including OS, logical volume manager, file system, and device driver are also explained. Virtualization at application and compute is also discussed in the lesson.

Application

- A software program that provides logic for computing operations
- Commonly deployed applications in a data center
 - ▶ Business applications – email, enterprise resource planning (ERP), decision support system (DSS)
 - ▶ Management applications – resource management, performance tuning, virtualization
 - ▶ Data protection applications – backup, replication
 - ▶ Security applications – authentication, antivirus

An *application* is a computer program that provides the logic for computing operations. The application sends requests to the underlying operating system to perform read/write (R/W) operations on the storage devices. Applications can be layered on the database, which in turn uses the OS services to perform R/W operations on the storage devices.

Applications deployed in a data center environment are commonly categorized as business applications, infrastructure management applications, data protection applications, and security applications. Some examples of these applications are e-mail, enterprise resource planning (ERP), decision support system (DSS), resource management, backup, authentication and antivirus applications, and so on.

Key I/O Characteristics of an Application

- I/O read-write ratio
- Sequential versus random
- I/O size

The characteristics of I/Os generated by the application influence the overall performance of storage system and storage solution designs. The common I/O characteristics of an application are I/O read-write ratio, sequential versus random, and I/O size. For more information on these characteristics, refer to the PDF on Application I/O characteristics.

Application Virtualization

Application Virtualization

It is the technique of presenting an application to an end user without any installation, integration, or dependencies on the underlying computing platform.

- Allows application to be delivered in an isolated environment
 - ▶ Aggregates Operating System (OS) resources and the application into a virtualized container
 - ▶ Ensures integrity of Operating System (OS) and applications
 - ▶ Avoids conflicts between different applications or different versions of the same application

Application virtualization breaks the dependency between the application and the underlying platform (OS and hardware). Application virtualization encapsulates the application and the required OS resources within a virtualized container. This technology provides the ability to deploy applications without making any change to the underlying OS, file system, or registry of the computing platform on which they are deployed. Because virtualized applications run in an isolated environment, the underlying OS and other applications are protected from potential corruptions. There are many scenarios in which conflicts might arise if multiple applications or multiple versions of the same application are installed on the same computing platform. Application virtualization eliminates this conflict by isolating different versions of an application and the associated O/S resources.

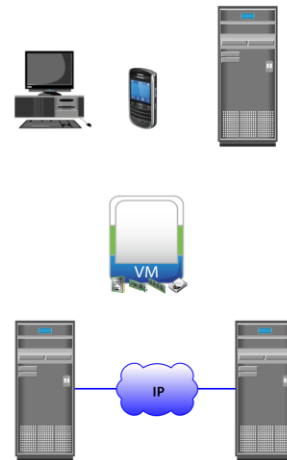
Database Management System (DBMS)

- Database is a structured way to store data in logically organized tables that are interrelated
 - ▶ Helps to optimize the storage and retrieval of data
- DBMS controls the creation, maintenance, and use of databases
 - ▶ Processes an application's request for data
 - ▶ Instructs the OS to retrieve the appropriate data from storage
- Popular DBMS examples are MySQL, Oracle RDBMS, SQL Server, etc.

A database is a structured way to store data in logically organized tables that are interrelated. A database helps to optimize the storage and retrieval of data. A DBMS controls the creation, maintenance, and use of a database. The DBMS processes an application's request for data and instructs the operating system to transfer the appropriate data from the storage.

Host (Compute)

- Resource that runs applications with the help of underlying computing components
 - ▶ Example: Servers, mainframes, laptop, desktops, tablets, server clusters, etc.
- Consists of hardware and software components
- Hardware components
 - ▶ Include CPU, memory, and input/output (I/O) devices
- Software components
 - ▶ Include OS, device driver, file system, volume manager, and so on



Users store and retrieve data through applications. The computers on which these applications run are referred to as *hosts* or *compute systems*. Hosts can be physical or virtual machines. A compute virtualization software enables creating virtual machines on top of physical compute infrastructure. Compute virtualization and virtual machines are discussed later in this module.

Examples of physical hosts include desktop computers, servers or a cluster of servers, virtual servers, laptops, and mobile devices. A host consists of CPU, memory, I/O devices, and a collection of software to perform computing operations. This software includes the operating system, file system, logical volume manager, device drivers, and so on. These software can be installed individually or may be part of the operating system.

Operating Systems

- In a traditional environment OS resides between the applications and the hardware
 - ▶ Responsible for controlling the environment
- In a virtualized environment virtualization layer works between OS and hardware
 - ▶ Virtualization layer controls the environment
 - ▶ OS works as a guest and only controls the application environment
 - ▶ In some implementation OS is modified to communicate with virtualization layer

In a traditional computing environment, an *operating system* controls all the aspects of computing. It works between the application and physical components of a compute system. One of the services it provides to the application is data access. The operating system also monitors and responds to user actions and the environment. It organizes and controls hardware components and manages the allocation of hardware resources. It provides basic security for the access and usage of all managed resources. An operating system also performs basic storage management tasks while managing other underlying components, such as the file system, volume manager, and device drivers.

In a virtualized compute environment, the virtualization layer works between the operating system and the hardware resources. Here, the OS might work differently based on the type of the compute virtualization implemented. In a typical implementation, the OS works as a guest and performs only the activities related to application interaction. In this case, hardware management functions are handled by the virtualization layer.

Memory Virtualization

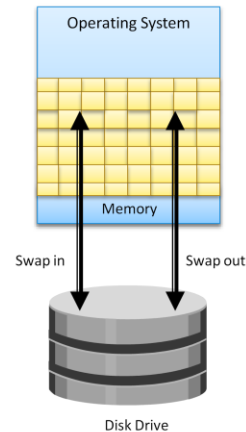
- An OS feature that presents larger memory to the application than physically available
- Additional memory space comes from disk storage
- Provides efficient use of available physical memory

Memory has been, and continues to be, an expensive component of a host. It determines both the size and number of applications that can run on a host. *Memory virtualization* enables multiple applications and processes, whose aggregate memory requirement is greater than the available physical memory, to run on a host without impacting each other.

Memory virtualization is an operating system feature that virtualizes the physical memory (RAM) of a host. It creates a virtual memory with an address space larger than the physical memory space present in the compute system. The virtual memory encompasses the address space of the physical memory and part of the disk storage. The operating system utility that manages the virtual memory is known as the *virtual memory manager* (VMM). The VMM manages the virtual-to-physical memory mapping and fetches data from the disk storage when a process references a virtual address that points to data at the disk storage. The space used by the VMM on the disk is known as a swap space.

Swap Space

- Space used on the disk for virtual memory is called 'swap space/swap file or page file'
- Inactive memory pages are moved from physical memory to the swap file
- Data access from swap file is slower – use of flash drives for swap space gives best performance



A *swap space* (also known as *page file* or *swap file*) is a portion of the disk drive that appears like physical memory to the operating system.

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 operating system typically moves the least used pages into the swap file so that enough RAM is available for processes that are more active. Access to swap file pages is slower than physical memory pages because swap file pages are allocated on the disk drive which is slower than physical memory.

Logical Volume Manager (LVM)

- Responsible for creating and controlling host level logical storage
- Physical view of storage is converted to a logical view
- Logical data blocks are mapped to physical data blocks

In the early days, entire disk drive would be allocated to the file system or other data entity used by the operating system or application. The disadvantage was lack of flexibility. When a disk drive ran out of space, there was no easy way to extend the file system's size. Also, as the storage capacity of the disk drive increased, allocating the entire disk drive for the file system often resulted in underutilization of storage capacity.

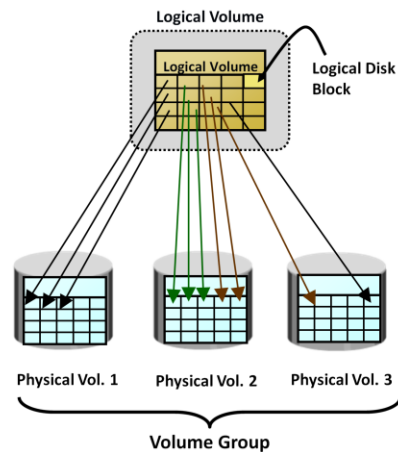
The evolution of *Logical Volume Managers* (LVMs) enabled dynamic extension of file system capacity and efficient storage management. LVM is software that runs on the compute system and manages logical and physical storage. LVM is an intermediate layer between the file system and the physical disk. It can partition a larger-capacity disk into virtual, smaller-capacity volumes (the process is called *partitioning*) or aggregate several smaller disks to form a larger virtual volume. (The process is called *concatenation*).

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.

Today, logical volume managers are mostly offered as part of the operating system.

Logical Volume Manager Components

- One or more Physical Volumes form a Volume Group
 - ▶ LVM manages Volume Groups as a single entity
- Logical volumes are created from the volume group



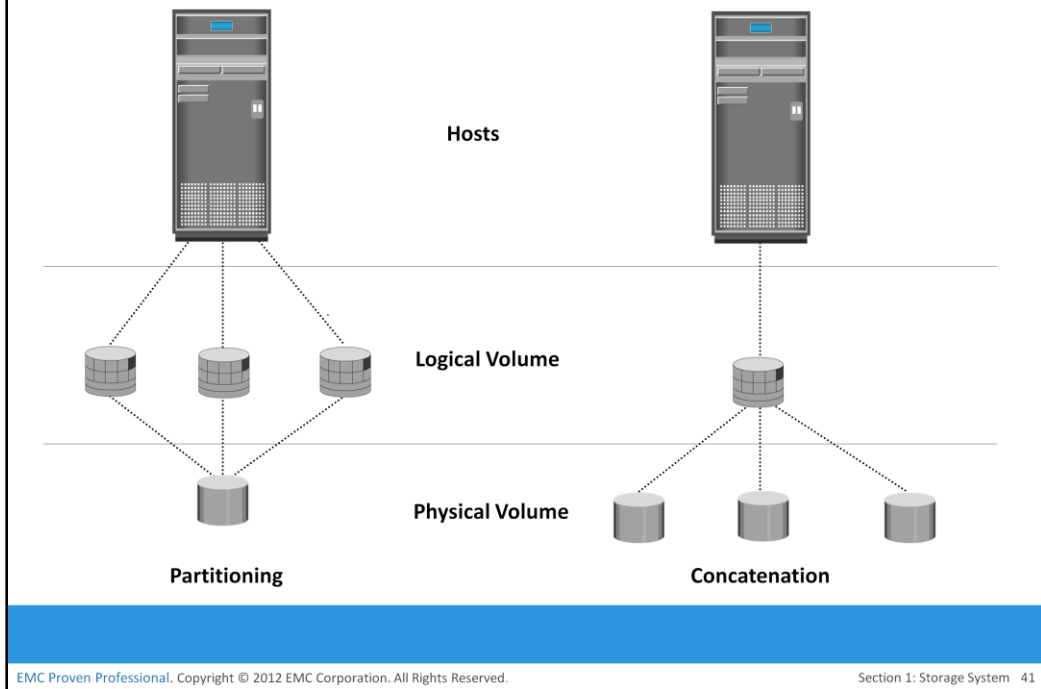
EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 40

The basic LVM components are physical volumes, volume groups, and logical volumes. In LVM terminology, each physical disk connected to the host system is a *physical volume* (PV). A *volume group* is created by grouping together one or more physical volumes. A unique *physical volume identifier* (PVID) is assigned to each physical volume 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 physical volume becomes part of a volume group. Each physical volume 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 physical extents. The LV appears as a physical device to the operating system. 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.

LVM Example: Partitioning and Concatenation



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* (LVs). 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.

File System – 1

- File is a collection of related records or data stored as a unit
- File system is hierarchical structure of files
 - ▶ Examples: FAT 32, NTFS, UNIX FS, and EXT2/3
- File system 'block' is the smallest unit allocated for storing data
- Metadata is a data about data
 - ▶ File name, size, location, date of creation/modification, owner, permissions, etc.

A *file* is a collection of related records or data stored as a unit with a name. A *file system* is a hierarchical structure of files. A file system enables easy access to data files residing within a disk drive, a disk partition, or a logical volume. A file system consists of logical structures and software routines that control access to files. It provides users with the functionality to create, modify, delete, and access files. Access to files on the disks is controlled by the permissions assigned to the file by the owner, which are also maintained by the file system.

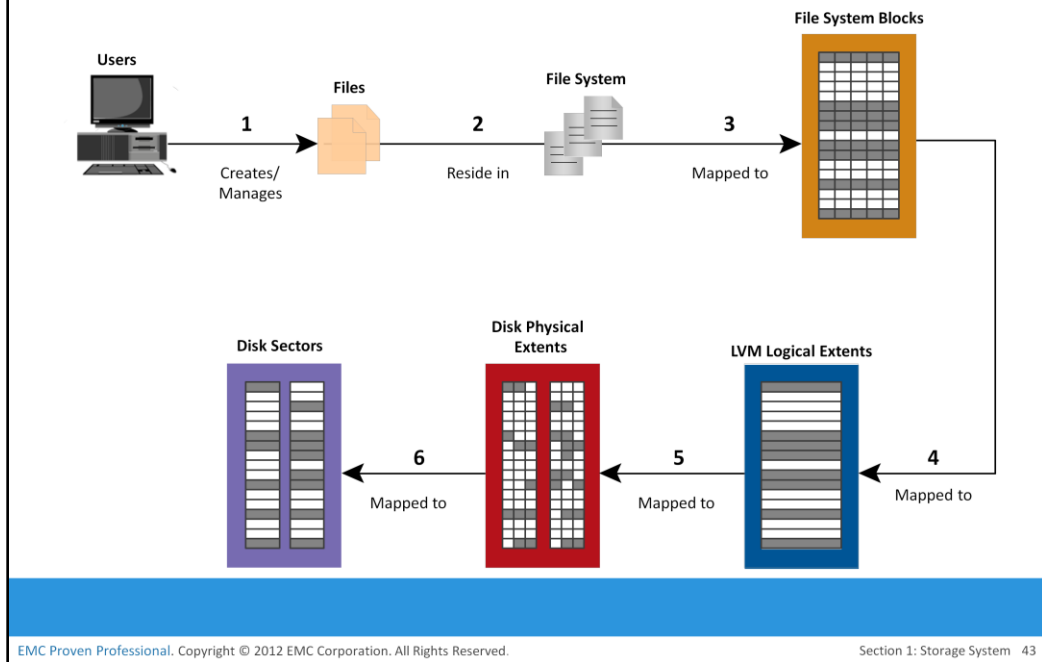
A file system organizes data in a structured hierarchical manner via the use of directories, which are containers for storing pointers to multiple files. All file systems maintain a pointer map to the directories, subdirectories, and files that are part of the file system.

A file system *block* is the smallest unit allocated for storing data. Each file system block is a contiguous area on the physical disk. The block size of a file system is fixed at the time of its creation. The file system size depends on the block size and the total number of file system blocks. A file can span multiple file system blocks because most files are larger than the predefined block size of the file system. File system blocks cease to be contiguous and become fragmented when new blocks are added or deleted. Over time, as files grow larger, the file system becomes increasingly fragmented.

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 file system to be considered healthy.

Examples of some common file systems are FAT 32 (File Allocation Table) and NT File System (NTFS) for Microsoft Windows, UNIX File System (UFS) and Extended File System (EXT2/3) for Linux.

File System – 2



The following list shows the process of mapping user files to the disk 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 disk physical extents either by the operating system or by the LVM.
6. These physical extents are mapped to the disk 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 disk sectors.

Device Driver

- It is a software that enables the OS to recognize the specific device

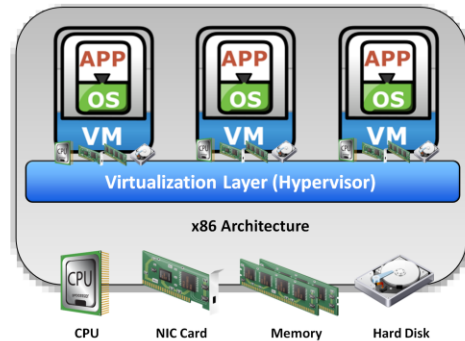
A *device driver* is a special software that permits the operating system to interact with a specific device such as a printer, a mouse, or a disk drive. A device driver enables the operating system to recognize the device and to access and control devices. Device drivers are hardware-dependent and operating-system-specific.

Compute Virtualization

Compute Virtualization

It is a technique of masking or abstracting the physical compute hardware and enabling multiple operating systems (OSs) to run concurrently on a single or clustered physical machine(s).

- Enables creation of multiple virtual machines (VMs), each running an OS and application
 - ▶ VM is a logical entity that looks and behaves like physical machine
- Virtualization layer resides between hardware and VMs
 - ▶ Also known as hypervisor
- VMs are provided with standardized hardware resources



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

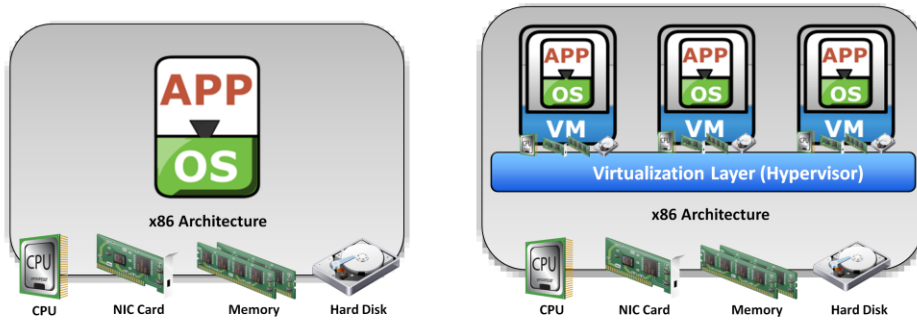
Section 1: Storage System 45

Compute virtualization is a technique of masking or abstracting the physical hardware from the operating system. It enables multiple operating systems to run concurrently on a single or clustered physical machine(s). This technique enables creating portable virtual compute systems called *virtual machines (VMs)*. Each VM runs an operating system and application instance in an isolated manner.

Compute virtualization is achieved by a virtualization layer that resides between the hardware and virtual machines. This layer is also called the *hypervisor*. The hypervisor provides hardware resources, such as CPU, memory, and network to all the virtual machines. Within a physical server, a large number of virtual machines can be created depending on the hardware capabilities of the physical server.

A virtual machine is a logical entity but appears like a physical host to the operating system, with its own CPU, memory, network controller, and disks. However, all VMs share the same underlying physical hardware in an isolated manner. From a hypervisor perspective, virtual machines are discrete sets of files that include VM configuration file, data files, and so on.

Need for Compute Virtualization



Before Virtualization	After Virtualization
• Runs single operating system (OS) per machine at a time • Couples s/w and h/w tightly • May create conflicts when multiple applications run on the same machine • Underutilizes resources • Is inflexible and expensive	• Runs multiple operating systems (OSs) per physical machine concurrently • Makes OS and applications h/w independent • Isolates VM from each other, hence, no conflict • Improves resource utilization • Offers flexible infrastructure at low cost

EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 46

Typically, a physical server often faces resource-conflict issues when two or more applications running on the server have conflicting requirements. For example, applications might need different values in the same registry entry, different versions of the same DLL, and so on. These issues are further compounded with an application's high-availability requirements. As a result, the servers are limited to serve only one application at a time. This causes organizations to purchase new physical machines for every application they deploy, resulting in expensive and inflexible infrastructure. On the other hand, many applications do not take full advantage of the hardware capabilities available to them. Consequently, resources such as processors, memory, and storage remain underutilized.

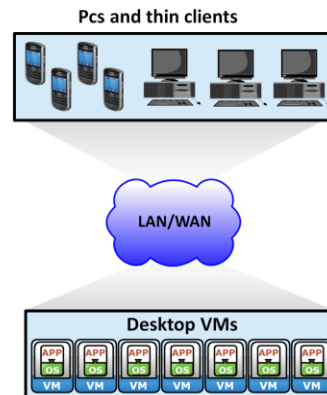
Compute virtualization enables to overcome these challenges by allowing multiple operating systems and applications to run on a single physical machine. This technique significantly improves server utilization and provides server consolidation. Server consolidation enables organizations to run their data center with fewer servers. This, in turn, cuts down the cost of new server acquisition, reduces operational cost, and saves data center floor and rack space. Creation of VMs takes less time compared to a physical server setup; organizations can provision servers faster and with ease. Individual VMs can be restarted, upgraded, or even crashed, without affecting the other VMs on the same physical machine. Moreover, VMs can be copied or moved from one physical machine to another without causing application downtime.

Desktop Virtualization

Desktop Virtualization

It is a technology which enables detachment of the user state, the Operating System (OS), and the applications from endpoint devices.

- Enables organizations to host and centrally manage desktops
 - ▶ Desktops run as virtual machines within the data center and accessed over a network
- Desktop virtualization benefits
 - ▶ Flexibility of access due to enablement of thin clients
 - ▶ Improved data security
 - ▶ Simplified data backup and PC maintenance



With the traditional desktop, the OS, applications, and user profiles are all tied to a specific piece of hardware. With legacy desktops, business productivity is impacted greatly when a client device is broken or lost. *Desktop virtualization* breaks the dependency between the hardware and its OS, applications, user profiles, and settings. This enables the IT staff to change, update, and deploy these elements independently. Desktops hosted at the data center and runs on virtual machines, whereas users remotely access these desktops from a variety of client devices, such as laptop, desktop, and mobile devices (also called Thin devices). Application execution and data storage are performed centrally at the data center instead of at the client devices. Because desktops run as virtual machines within an organization's data center, it mitigates the risk of data leakage and theft. It also helps to perform centralized backup and simplifies compliance procedures. Virtual desktops are easy to maintain because it is simple to apply patches, deploy new applications and OS, and provision or remove users centrally.

Module 2: Data Center Environment

Lesson 2: Connectivity

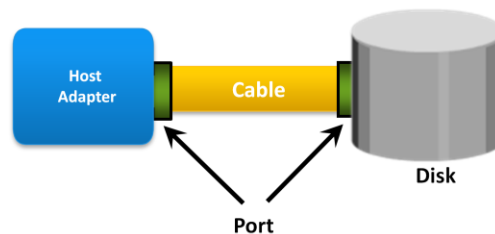
During this lesson the following topics are covered:

- Physical components of connectivity
- Storage connectivity protocols

This lesson covers physical components of connectivity and storage connectivity protocols. These protocols include IDE/ATA, SCSI, Fibre Channel and IP.

Connectivity

- Interconnection between hosts or between a host and peripheral devices, such as storage
- Physical Components of Connectivity are:
 - ▶ Host interface card, port, and cable
- Protocol = a defined format for communication between sending and receiving devices
 - ▶ Popular storage interface protocols: IDE/ATA and SCSI



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 49

Connectivity refers to the interconnection between hosts or between a host and peripheral devices, such as printers or storage devices. The discussion here focuses only on the connectivity between the host and the storage device. Connectivity and communication between host and storage are enabled using physical components and interface protocols.

The *physical components* of connectivity are the hardware elements that connect the host to storage. Three physical components of connectivity between the host and storage are host interface device, port, and cable

A *host interface device* or *host adapter* connects a host to other hosts and storage devices. Examples of host interface devices are host bus adapter (HBA) and network interface card (NIC). *Host bus adaptor (HBA)* is an application-specific integrated circuit (ASIC) board that performs I/O interface functions between the host and storage, relieving the CPU from additional I/O processing workload. A host typically contains multiple HBAs.

A *port* is a specialized outlet that enables connectivity between the host and external devices. An HBA may contain one or more ports to connect the host to the storage device. *Cables* connect hosts to internal or external devices using copper or fiber optic media.

A *protocol* enables communication between the host and storage. Protocols are implemented using interface devices (or controllers) at both source and destination. The popular interface protocols used for host to storage communications are *Integrated Device Electronics/Advanced Technology Attachment (IDE/ATA)*, *Small Computer System Interface (SCSI)*, *Fibre Channel (FC)*, and *Internet Protocol (IP)*.

IDE/ATA and Serial ATA

- Integrated Device Electronics (IDE)/Advanced Technology Attachment (ATA)
 - ▶ Popular interface used to connect hard disks or CD-ROM drives
 - ▶ Available with variety of standards and names
- Serial Advanced Technology Attachment (SATA)
 - ▶ Serial version of the IDE/ATA specification that has replaced the parallel ATA
 - ▶ Inexpensive storage interconnect, typically used for internal connectivity
 - ▶ Provides data transfer rate up to 6 Gb/s (standard 3.0)

IDE/ATA is a popular interface protocol standard used for connecting storage devices, such as disk drives and CD-ROM 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 per second. 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 3.0 provides a data transfer rate up to 6 Gb/s.

SCSI and SAS

- Parallel Small computer system interface (SCSI)
 - ▶ Popular standard for connecting host and peripheral devices
 - ▶▶ Commonly used for storage connectivity in servers
 - ▶ Higher cost than IDE/ATA, therefore not popular in PC environments
 - ▶ Available in wide variety of related technologies and standards
 - ▶ Support up to 16 devices on a single bus
 - ▶ Ultra-640 version provides data transfer speed up to 640 MB/s
- Serial Attached SCSI (SAS)
 - ▶ Point-to-point serial protocol replacing parallel SCSI
 - ▶ Supports data transfer rate up to 6 Gb/s (SAS 2.0)

SCSI has emerged as a preferred connectivity protocol in high-end computers. This protocol supports parallel transmission and offers improved performance, scalability, and compatibility compared to ATA. However, the high cost associated with SCSI limits its popularity among home or personal desktop users. Over the years, SCSI has been enhanced and now includes a wide variety of related technologies and standards. SCSI supports up to 16 devices on a single bus and provides data transfer rates up to 640 MB/s (for the Ultra-640 version).

Serial attached SCSI (SAS) is a point-to-point serial protocol that provides an alternative to parallel SCSI. A newer version (SAS 2.0) of serial SCSI supports a data transfer rate up to 6 Gb/s.

Fibre Channel and IP

- Fibre Channel (FC)
 - ▶ Widely used protocol for high speed communication to the storage device
 - ▶ 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
- Internet Protocol (IP)
 - ▶ Traditionally used to transfer host-to-host traffic
 - ▶ Provide opportunity to leverage existing IP based network for storage communication
 - ▶▶ Examples: iSCSI and FCIP protocols

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 5.

IP is a network protocol that has been traditionally used for host-to-host traffic. With the emergence of new technologies, an IP network has become a viable option for host-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 host-to-storage communication. These protocols are detailed in module 6.

Module 2: Data Center Environment

Lesson 3: Storage

During this lesson the following topics are covered:

- Various storage options
- Disk drive components, addressing, and performance
- Enterprise Flash drives
- Host access to storage and direct-attached storage

This lesson covers the most important element of a data center – Storage. Various storage medias and options are discussed with focus on disk drives. Components, structure, addressing, and factors that impacts disk drives performance are detailed in the lesson. Further it covers new generation flash drives and their benefits. Finally it introduces various methods of accessing storage from the host with details of direct-attached storage options.

Storage Options

- Tapes
- CD/DVD
- Disk Drives – Mechanical and Flash based solid-state drives

The storage is a core component in a data center. A storage device uses magnetic, optic, or solid state media. Disks, tapes, and diskettes use magnetic media, whereas CD/DVD uses optical media for storage. Removable Flash memory or Flash drives are examples of solid state media.

Magnetic Tapes

- Low cost solution for long term data storage
 - ▶ Preferred option for backup destination in the past
- Limitations
 - ▶ Sequential data access
 - ▶ Single application access at a time
 - ▶ Physical wear and tear
 - ▶ Storage/retrieval overheads

In the past, *tapes* were the most popular storage option for backups because of their low cost. However, tapes have various limitations in terms of performance and management as listed here:

- Data is stored on the tape linearly along the length of the tape. Search and retrieval of data are done sequentially, and it invariably takes several seconds to access the data. As a result, random data access is slow and time-consuming. This limits tapes as a viable option for applications that require real-time, rapid access to data.
- In a shared computing environment, data stored on tape cannot be accessed by multiple applications simultaneously, restricting its use to one application at a time.
- On a tape drive, the read/write head touches the tape surface, so the tape degrades or wears out after repeated use.
- The storage and retrieval requirements of data from the tape and the overhead associated with managing the tape media are significant.

Due to these limitations and availability of low-cost disk drives, tapes are no longer a preferred choice as a backup destination for enterprise-class data centers.

Optical Discs

- Popularly used as distribution medium in small, single-user computing environments
- Limited in capacity and speed
- Write once and read many (WORM): CD-ROM, DVD-ROM
- Other variations: CD-RW, Blu-ray discs

Optical disc storage is popular in small, single-user computing environments. It is frequently used by individuals to store photos or as a backup medium on personal or laptop computers. It is also used as a distribution medium for small applications such as games, or as a means to transfer small amounts of data from one computer to another.

Optical discs have limited capacity and speed, which limit the use of optical media as a business data storage solution. The capability to write once and read many (WORM) is one advantage of optical disc storage. A CD-ROM is an example of a WORM device. Optical discs, to some degree, guarantee that the content has not been altered. Therefore, it can be used as a low-cost alternative for long-term storage of relatively small amounts of fixed content that do not change after it is created. Collections of optical discs in an array, called a *jukebox*, are still used as a fixed-content storage solution. Other forms of optical discs include CD-RW, Blu-ray disc, and other variations of DVD.

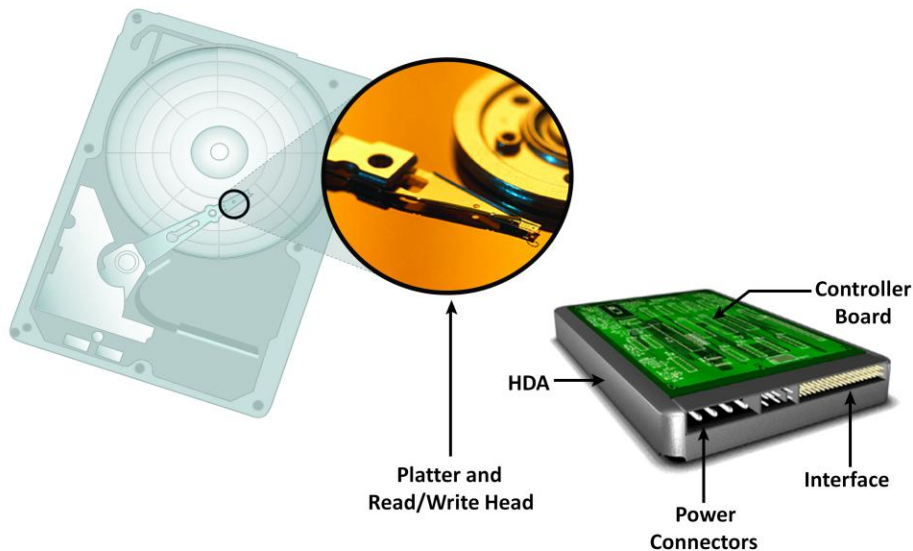
Disk Drives

- Mechanical disk drive
 - ▶ Most popular storage medium
 - ▶ Large storage capacity
 - ▶ Random read/write access
- Flash drives
 - ▶ Uses semiconductor media
 - ▶ Provide high performance and low power consumption

Disk drives are the most popular storage medium used in modern computers for storing and accessing data for performance-intensive, online applications. Disks support rapid access to random data locations. This means that data can be written or retrieved quickly for a large number of simultaneous users or applications. In addition, disks have a large capacity. Disk storage arrays are configured with multiple disks to provide increased capacity and enhanced performance. Flash drives (or solid stated drives - SSDs) use semiconductor media and provides high performance and low power consumption.

Following slides detail various components and performance characteristics of disk drives. Flash drives are detailed later in the module.

Disk Drive Components



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 58

The key components of a hard disk drive are platter, spindle, read-write head, actuator arm assembly, and controller board. I/O operations in a HDD is performed by rapidly moving the arm across the rotating flat platters coated with magnetic particles. Data is transferred between the disk controller and magnetic platters through the read-write (R/W) head which is attached to the arm. Data can be recorded and erased on magnetic platters any number of times.

Platter: A typical HDD consists of one or more flat circular disks called *platters*. The data is recorded on these platters in binary codes (0s and 1s). The set of rotating platters is sealed in a case, called *Head Disk Assembly* (HDA). A platter is a rigid, round disk coated with magnetic material on both surfaces (top and bottom). The data is encoded by polarizing the magnetic area, or domains, of the disk surface. Data can be written to or read from both surfaces of the platter. The number of platters and the storage capacity of each platter determine the total capacity of the drive.

Spindle: A spindle connects all the platters and is connected to a motor. The motor of the spindle rotates with a constant speed. The disk platter spins at a speed of several thousands of revolutions per minute (rpm). Common spindle speeds are 5,400 rpm, 7,200 rpm, 10,000 rpm, and 15,000 rpm. The speed of the platter is increasing with improvements in technology; although, the extent to which it can be improved is limited.

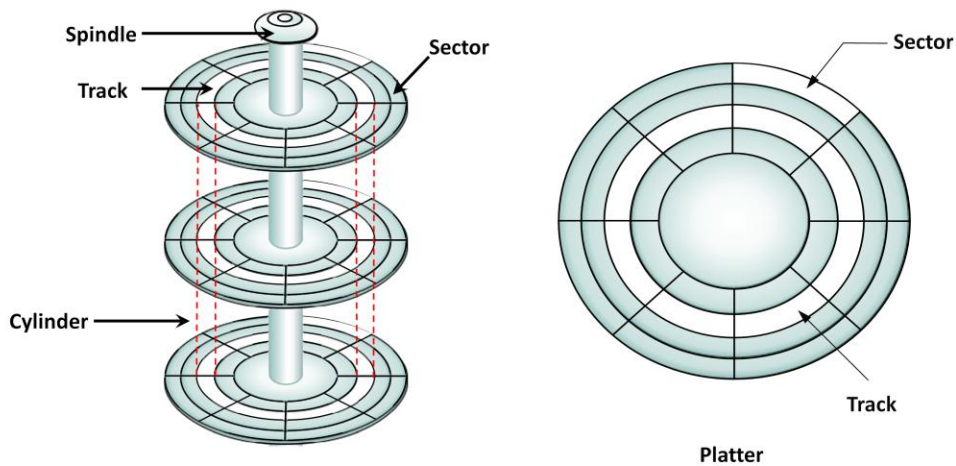
Cont...

Read/Write Head: *Read/Write (R/W) heads*, read and write data from or to platters. Drives have two R/W heads per platter, one for each surface of the platter. The R/W head changes the magnetic polarization on the surface of the platter when writing data. While reading data, the head detects the magnetic polarization on the surface of the platter. During reads and writes, the R/W head senses the magnetic polarization and never touches the surface of the platter. When the spindle is rotating, there is a microscopic air gap maintained between the R/W heads and the platters, known as the *head flying height*. This air gap is removed when the spindle stops rotating and the R/W head rests on a special area on the platter near the spindle. This area is called the *landing zone*. The landing zone is coated with a lubricant to reduce friction between the head and the platter. The logic on the disk drive ensures that heads are moved to the landing zone before they touch the surface. If the drive malfunctions and the R/W head accidentally touches the surface of the platter outside the landing zone, a *head crash* occurs. In a head crash, the magnetic coating on the platter is scratched and may cause damage to the R/W head. A head crash generally results in data loss.

Actuator Arm Assembly: R/W heads are mounted on the *actuator arm assembly*, which positions the R/W head at the location on the platter where the data needs to be written or read. The R/W heads for all platters on a drive are attached to one actuator arm assembly and move across the platters simultaneously.

Drive Controller Board: The controller is a printed circuit board, mounted at the bottom of a disk drive. It consists of a microprocessor, internal memory, circuitry, and firmware. The firmware controls the power to the spindle motor and the speed of the motor. It also manages the communication between the drive and the host. In addition, it controls the R/W operations by moving the actuator arm and switching between different R/W heads, and performs the optimization of data access.

Disk Structure



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 60

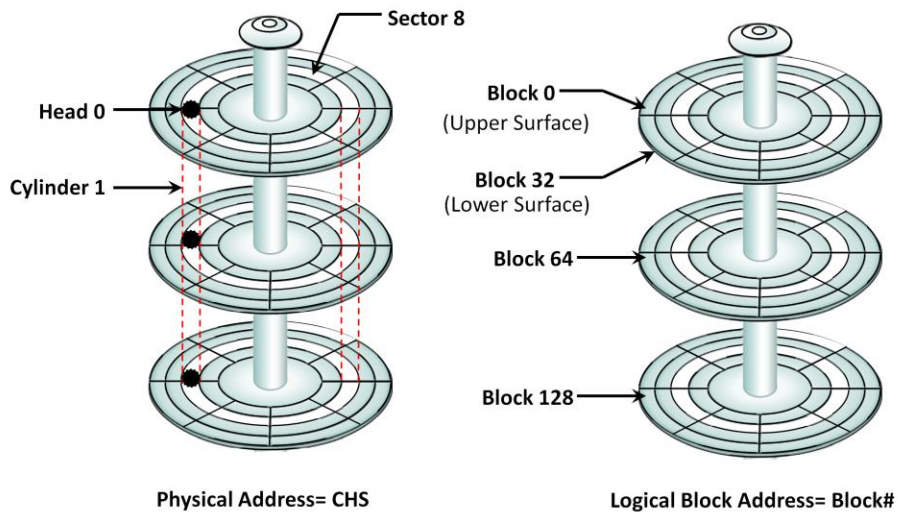
Data on the disk is recorded on *tracks*, which are concentric rings on the platter around the spindle. The tracks are numbered, starting from zero, from the outer edge of the platter. The number of *tracks per inch* (TPI) on the platter (or the *track density*) measures how tightly the tracks are packed on a platter.

Each track is divided into smaller units called *sectors*. A sector is the smallest, individually addressable unit of storage. The track and sector structure is written on the platter by the drive manufacturer using a low-level formatting operation. The number of sectors per track varies according to the drive type. The first personal computer disks had 17 sectors per track. Recent disks have a much larger number of sectors on a single track. There can be thousands of tracks on a platter, depending on the physical dimensions and recording density of the platter.

Typically, a sector holds 512 bytes of user data; although, some disks can be formatted with larger sector sizes. In addition to user data, a sector also stores other information, such as the sector number, head number or platter number, and track number. This information helps the controller to locate the data on the drive.

A cylinder is a set of identical tracks on both surfaces of each drive platter. The location of R/W heads is referred to by the cylinder number, not by the track number.

Logical Block Addressing



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 61

Earlier drives used physical addresses consisting of the *cylinder, head, & sector (CHS)* number to refer to specific locations on the disk, and the host operating system had to be aware of the geometry of each disk used. *Logical block addressing (LBA)* has simplified the addressing by using a linear address to access physical blocks of data. The disk controller translates LBA to a CHS address, and the host needs to know only the size of the disk drive in terms of the number of blocks. The logical blocks are mapped to physical sectors on a 1:1 basis.

In the slide, the drive shows eight sectors per track, six heads, and four cylinders. This means a total of $8 \times 6 \times 4 = 192$ blocks, so the block number ranges from 0 to 191. Each block has its own unique address.

Assuming that the sector holds 512 bytes, a 500-GB drive with a formatted capacity of 465.7 GB has in excess of 976,000,000 blocks.

Disk Drive Performance

- Electromechanical device
 - ▶ Impacts the overall performance of the storage system
- Disk service time
 - ▶ Time taken by a disk to complete an I/O request, depends on:
 - ▶▶ Seek time
 - ▶▶ Rotational latency
 - ▶▶ Data transfer rate

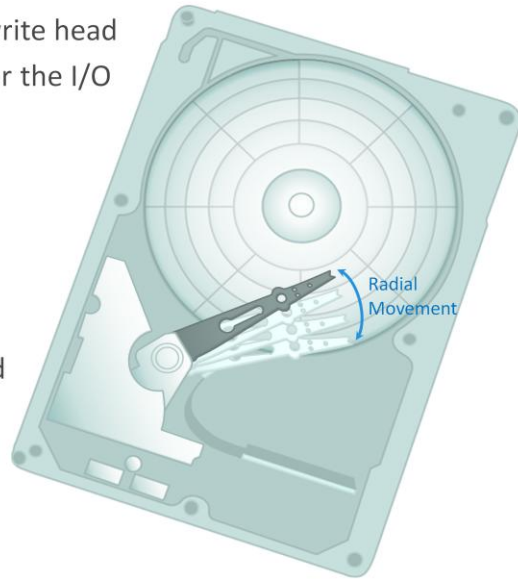
Disk service time = seek time + rotational latency + data transfer time

A disk drive is an electromechanical device that governs the overall performance of the storage system environment. The various factors that affect the performance of disk drives are:

- Seek time
- Rotational latency
- Data transfer rate

Seek Time

- Time taken to position the read/write head
- The lower the seek time, the faster the I/O operation
- Seek time specifications include
 - ▶ Full stroke
 - ▶ Average
 - ▶ Track-to-track
- The seek time of a disk is specified by the drive manufacturer



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 63

The *seek time* (also called *access time*) describes the time taken to position the R/W heads across the platter with a radial movement (moving along the radius of the platter). In other words, it is the time taken to position and settle the arm and the head over the correct track. Therefore, the lower the seek time, the faster the I/O operation. Disk vendors publish the following seek time specifications:

Full Stroke: The time taken by the R/W head to move across the entire width of the disk, from the innermost track to the outermost track.

Average: The average time taken by the R/W head to move from one random track to another, normally listed as the time for one-third of a full stroke.

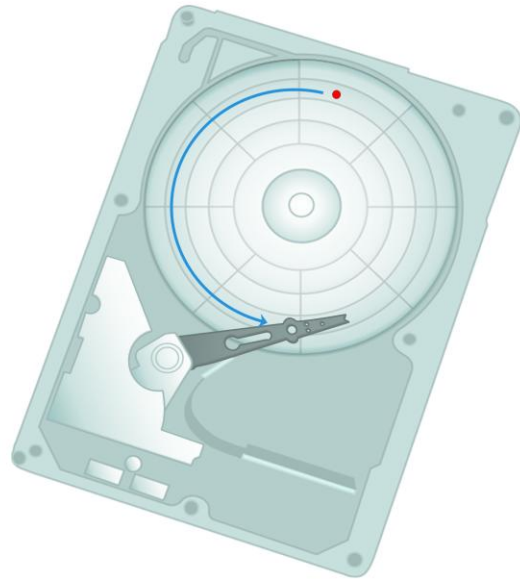
Track-to-Track: The time taken by the R/W head to move between adjacent tracks.

Each of these specifications is measured in milliseconds. The seek time of a disk is typically specified by the drive manufacturer. The average seek time on a modern disk is typically in the range of 3 to 15 milliseconds. Seek time has more impact on the I/O operation of random tracks rather than the adjacent tracks. To minimize the seek time, data can be written to only a subset of the available cylinders. This results in lower usable capacity than the actual capacity of the drive. For example, a 500-GB disk drive is set up to use only the first 40 percent of the cylinders and is effectively treated as a 200-GB drive. This is known as *short-stroking* the drive.

Rotational Latency

- The time taken by the platter to rotate and position the data under the R/W head
- Depends on the rotation speed of the spindle
- Average rotational latency
 - ▶ One-half of the time taken for a full rotation
 - ▶ For 'X' rpm, drive latency is calculated in milliseconds as:

$$= \frac{1/2}{(X/60)}$$



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 64

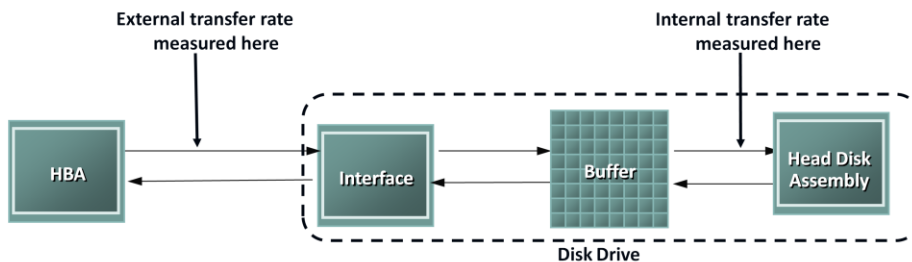
To access data, the actuator arm moves the R/W head over the platter to a particular track while the platter spins to position the requested sector under the R/W head. The time taken by the platter to rotate and position the data under the R/W head is called *rotational latency*. This latency depends on the rotation speed of the spindle and is measured in milliseconds. The average rotational latency is one-half of the time taken for a full rotation. Similar to the seek time, rotational latency has more impact on the reading/writing of random sectors on the disk than on the same operations on adjacent sectors.

Average rotational latency is approximately 5.5 ms for a 5,400-rpm drive, and around 2.0 ms for a 15,000-rpm (or 250-rps revolution per second) drive as shown here.

Av. rotational latency for 15K rpm or 250 rps (15000/60) drive is = $(1/2)/250=2$ milliseconds

Data Transfer Rate

- Average amount of data per unit time that the drive can deliver to the HBA
 - ▶ Internal transfer rate: Speed at which data moves from a platter's surface to the internal buffer of the disk
 - ▶ External transfer rate: Rate at which data move through the interface to the HBA



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 65

The *data transfer rate* (also called *transfer rate*) refers to the average amount of data per unit time that the drive can deliver to the HBA. In a *read operation*, the data first moves from disk platters to R/W heads; then it moves to the drive's internal *buffer*. Finally, data moves from the buffer through the interface to the host HBA. In a *write operation*, the data moves from the HBA to the internal buffer of the disk drive through the drive's interface. The data then moves from the buffer to the R/W heads. Finally, it moves from the R/W heads to the platters. The data transfer rates during the R/W operations are measured in terms of internal and external transfer rates, as shown in the slide.

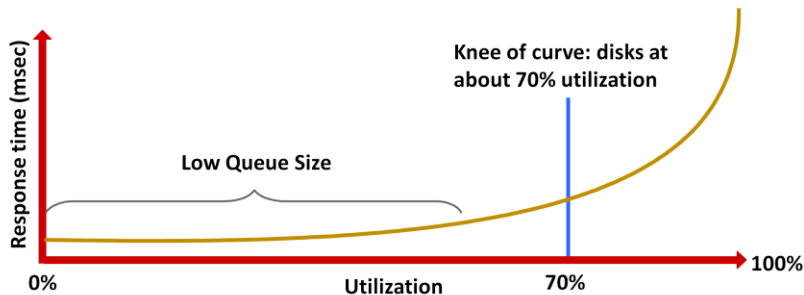
Internal transfer rate is the speed at which data moves from a platter's surface to the internal buffer (cache) of the disk. The internal transfer rate takes into account factors such as the seek time and rotational latency. *External transfer rate* is the rate at which data can move through the interface to the HBA. The external transfer rate is generally the advertised speed of the interface, such as 133 MB/s for ATA. The sustained external transfer rate is lower than the interface speed.

I/O Controller Utilization vs. Response Time

- Based on fundamental laws of disk drive performance:

$$Av. Response Time = \frac{Service Time}{(1 - Utilization)}$$

- ▶ Service time is time taken by the controller to serve an I/O
- For performance-sensitive applications, disks are commonly utilized below 70% of their I/O serving capability



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 66

Utilization of a disk I/O controller has a significant impact on the I/O response time. Consider that a disk is viewed as a black box consisting of two elements queue and disk I/O controller. *Queue* is the location where an I/O request waits before it is processed by the I/O controller and *disk I/O controller* processes I/Os waiting in the queue one by one.

The I/O requests arrive at the controller at the rate generated by the application. The I/O arrival rate, the queue length, and the time taken by the I/O controller to process each request determines the I/O response time. If the controller is busy or heavily utilized, the queue size will be large and the response time will be high. Based on the fundamental laws of disk drive performance, the relationship between controller utilization and average response time is given as:

$$Average response time = Service time / (1 - Utilization)$$

where, service time is the time taken by the controller to serve an I/O.

As the utilization reaches 100 percent that is, as the I/O controller saturates, the response time is closer to infinity. In essence, the saturated component, or the bottleneck, forces the serialization of I/O requests; meaning, each I/O request must wait for the completion of the I/O requests that preceded it. Figure in the slide shows a graph plotted between utilization and response time. The graph indicates that the response time changes are nonlinear as the utilization increases. When the average queue sizes are low, the response time remains low. The response time increases slowly with added load on the queue and increases exponentially when the utilization exceeds 70 percent. Therefore, for performance-sensitive applications, it is common to utilize disks below their 70 percent of I/O serving capability.

Storage Design Based on Application Requirements and Disk Drive Performance

- Disks required to meet an application's capacity need (D_C):

$$D_C = \frac{\text{Total capacity required}}{\text{Capacity of a single disk}}$$

- Disks required to meet application's performance need (D_P):

$$D_P = \frac{\text{IOPS generated by an application at peak workload}}{\text{IOPS serviced by single disk}}$$

- IOPS serviced by a disk (S) depends upon disk service time (T_S):

$$T_S = \text{Seek time} + \frac{0.5}{(\text{Disk rpm}/60)} + \frac{\text{Data block size}}{\text{Data transfer rate}}$$

- ▶ T_S is time taken for an I/O to complete, therefore IOPS serviced by a disk (S) is equal to $(1/T_S)$

$$\text{▶▶ For performance sensitive application } (S) = 0.7 \times \frac{1}{T_S}$$

$$\text{Disk required for an application} = \max(D_C, D_P)$$

Determining storage requirements for an application begins with determining the required storage capacity and I/O performance. Capacity can be easily estimated by the size and number of file systems and database components used by applications. The I/O size, I/O characteristics, and the number of I/Os generated by the application at peak workload are other factors that affect performance, I/O response time and design of storage system.

The disk service time (T_S) for an I/O is a key measure of disk performance; T_S , along with disk utilization rate (U), determines the I/O response time for an application. As discussed earlier the total disk service time is the sum of the seek time, rotational latency, and transfer time.

Note that transfer time is calculated based on the block size of the I/O and given data transfer rate of a disk drive—for example, an I/O with a block size of 32 KB and given disk data transfer rate 40MB/s; the transfer time will be 32 KB / 40 MB.

T_S determines the time taken by the I/O controller to serve an I/O, therefore, the maximum number of I/Os serviced per second or IOPS is $(1/T_S)$.

The IOPS calculated above represents the IOPS that can be achieved at potentially high levels of I/O controller utilization (close to 100 percent). If the application demands a faster response time, then the utilization for the disks should be maintained below 70 percent.

Based on this discussion, the total number of disks required for an application is computed as :

$$= \text{Max (Disks required for meeting capacity, Disks required for meeting performance)}$$

Cont..

Consider an example in which the capacity requirement for an application is 1.46 TB. The number of IOPS generated by the application at peak workload is estimated at 9,000 IOPS. The vendor specifies that a 146-GB, 15,000-rpm drive is capable of doing a maximum 180 IOPS.

In this example, the number of disks required to meet the capacity requirements will be $1.46 \text{ TB} / 146 \text{ GB} = 10$ disks.

To meet the application IOPS requirements, the number of disks required is $9,000 / 180 = 50$. However, if the application is response-time sensitive, the number of IOPS a disk drive can perform should be calculated based on 70-percent disk utilization. Considering this, the number of IOPS a disk can perform at 70 percent utilization is $180 \times 0.7 = 126$ IOPS. Therefore, the number of disks required to meet the application IOPS requirement will be $9,000 / 126 = 72$.

As a result, the number of disks required to meet the application requirements will be $\text{Max}(10, 72) = 72$ disks.

The preceding example indicates that from a capacity-perspective, 10 disks are sufficient; however, the number of disks required to meet application performance is 72. To optimize disk requirements from a performance perspective, various solutions are deployed in a real-time environment. Examples of these solutions are disk native command queuing, use of flash drives, RAID, and the use of cache memory. RAID and cache are detailed in module 3 and 4 respectively.

Enterprise Flash Drives

Conventional Hard Drives	Flash Drives
Mechanical delay due to seek time and rotational latency	Highest possible throughput per drive due to no mechanical movement
Limited performance and I/O serving capability	Very low latency per I/O and consistent I/O performance
More power consumption due to mechanical operations	High Energy efficiency • Lower power requirement per GB • Lower power requirement per IOPS
Low mean time between failure (MTBF)	High reliability due to no moving parts
Higher TCO due to more number of disks, power, cooling, and management cost	Overall less TCO

EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

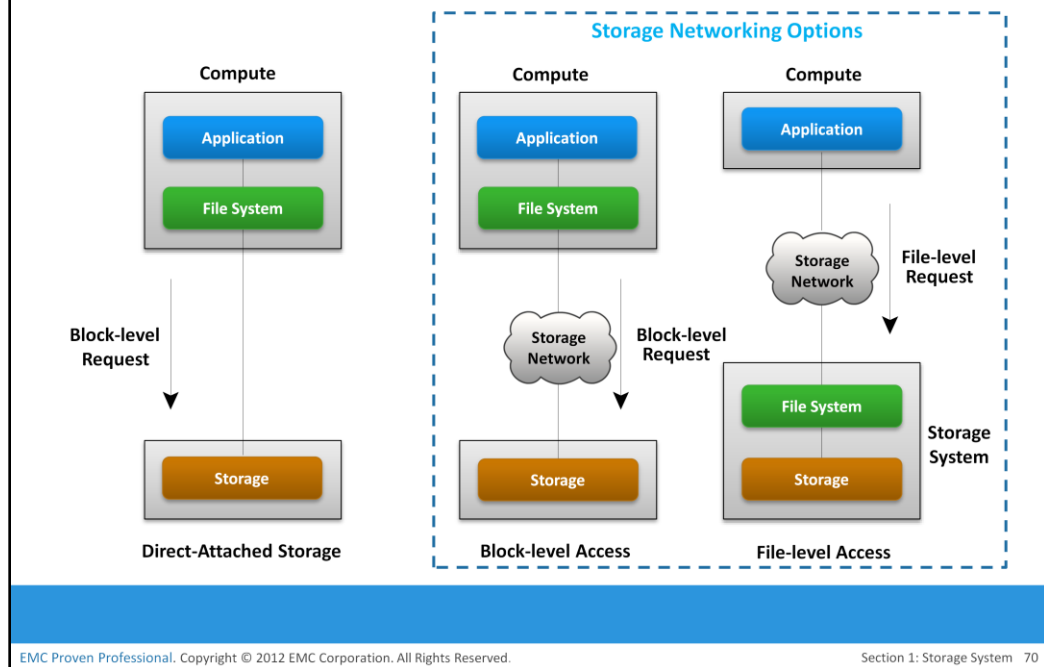
Section 1: Storage System 69

Traditionally, high I/O requirements of an application were met by simply using more disks. Availability of enterprise class flash drives (EFD) has changed the scenario.

Flash drives, also referred as solid state drives (SSDs), are new generation drives that deliver ultra-high performance required by performance-sensitive applications. Flash drives use semiconductor-based solid state memory (flash memory) to store and retrieve data. Unlike conventional mechanical disk drives, flash drives contain no moving parts; therefore, they do not have seek and rotational latencies. Flash drives deliver a high number of IOPS with very low response times. Also, being a semiconductor-based device, flash drives consume less power, compared to mechanical drives. Flash drives are especially suited for applications with small block size and random-read workloads that require consistently low (less than 1 ms) response times. Applications that need to process massive amounts of information quickly, such as currency exchange, electronic trading systems, and real-time data feed processing, benefit from flash drives.

Overall, flash drives provide better total cost of ownership (TCO) even though they cost more on \$/GB basis. By implementing flash drives, businesses can meet application performance requirements with far fewer drives (approximately 20 to 30 times less number of drives compared to conventional mechanical drives). This reduction not only provides savings in terms of drive cost, but also translates to savings for power, cooling, and space consumption. Fewer numbers of drives in the environment also means less cost for managing the storage.

Host Access to Storage



Data is accessed and stored by applications using the underlying infrastructure. The key components of this infrastructure are the operating system (or file system), connectivity, and storage. The storage device can be internal and (or) external to the host. In either case, the host controller card accesses the storage devices using predefined protocols, such as IDE/ATA, SCSI, or Fibre Channel (FC). IDE/ATA and SCSI are popularly used in small and personal computing environments for accessing internal storage. FC and iSCSI protocols are used for accessing data from an external storage device (or subsystems). External storage devices can be connected to the host directly or through the storage network. When the storage is connected directly to the host, it is referred as Direct-Attached Storage (DAS).

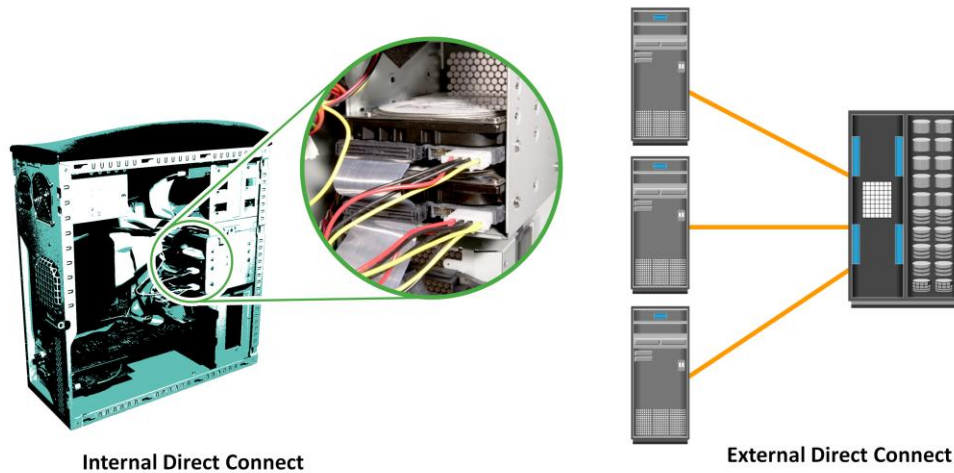
Data can be accessed over a network in one of the following ways: block level, file level, or object level. In general, the application requests data from the file system (or operating system) by specifying the filename and location. The file system maps the file attributes to the logical block address of the data and sends the request to the storage device. The storage device converts the logical block address (LBA) to a cylinder-head-sector (CHS) address and fetches the data.

In a block-level access, the file system is created on a host, and data is accessed on a network at the block level. In this case, raw disks or logical volumes are assigned to the host for creating the file system.

In a file-level access, the file system is created on a separate file server or at the storage side, and the file-level request is sent over a network. Because data is accessed at the file level, this method has higher overhead, as compared to the data accessed at the block level.

Object-level access is an intelligent evolution, whereby data is accessed over a network in terms of self-contained objects with a unique object identifier. Details of storage networking technologies and deployments are covered in later modules of this course.

Direct-Attached Storage (DAS)



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 71

DAS is an architecture in which storage is connected directly to the hosts. The internal disk drive of a host and the directly connected external storage array are examples of DAS. Although the implementation of storage networking technologies is gaining popularity, DAS has remained suitable for localized data access in a small environment, such as personal computing and workgroups. DAS is classified as internal or external, based on the location of the storage device with respect to the host.

In *internal DAS* architectures, the storage device is internally connected to the host by a serial or parallel bus. The physical bus has distance limitations and can be sustained only over a shorter distance for high-speed connectivity. In addition, most internal buses can support only a limited number of devices, and they occupy a large amount of space inside the host, making maintenance of other components difficult. In *external DAS* architectures, the host connects directly to the external storage device, and data is accessed at the block level. In most cases, communication between the host and the storage device takes place over a SCSI or FC protocol. Compared to internal DAS, an external DAS overcomes the distance limitations and provides centralized management of storage devices.

DAS Benefits and limitations: DAS requires a relatively lower initial investment than storage networking architectures. The DAS configuration is simple and can be deployed easily and rapidly. It requires fewer management tasks and less hardware and software elements to set up and operate. However, DAS does not scale well. A storage array has a limited number of ports, which restricts the number of hosts that can directly connect to the storage. Therefore, DAS does not make optimal use of resources and, moreover unused resources cannot be easily re-allocated, resulting in islands of over-utilized and under-utilized storage pools.

Module 2: Data Center Environment

Concept in Practice

- VMware ESXi

The concept in practice covers the product example of compute virtualization. It covers industry's leading hypervisor software VMware ESXi.

VMware ESXi

- Industry's leading hypervisor
 - ▶ Enable virtualization of x86 hardware platforms
- Physical machine that houses ESXi is called ESXi host
 - ▶ ESXi host abstracts physical compute resources to run multiple VMs concurrently on same physical server
- Two Components
 - ▶ VMKernel
 - ▶▶ Work similar to OS – responsible for process creation, resource scheduling, and so on
 - ▶ Virtual machine monitor
 - ▶▶ Performs binary translation for privileged OS instructions that can not be virtualized

VMware is the leading provider for server virtualization solution. VMware ESXi provides a platform called hypervisor. The hypervisor abstracts CPU, memory, and storage resources to run multiple virtual machines concurrently on the same physical server.

VMware ESXi is a hypervisor that installs on x86 hardware to enable server virtualization. It enables creating multiple virtual machines (VMs) that can run simultaneously on the same physical machine. A VM is a discrete set of files that can be moved, copied, and used as a template. All the files that make up a VM are typically stored in a single directory on a cluster file system called Virtual Machine File System (VMFS). The physical machine that houses ESXi is called ESXi host. The ESXi hosts provide physical resources used to run virtual machines. ESXi has two key components: VMkernel and Virtual Machine Monitor.

VMkernel provides functionality similar to that found in other operating systems, such as process creation, file system management, and process scheduling. It is designed to specifically support running multiple VMs and provide core functionality such as resource scheduling, I/O stacks, and so on.

The virtual machine monitor is responsible for executing commands on the CPUs and performing Binary Translation (BT). A virtual machine monitor performs hardware abstraction to appear as a physical machine with its own CPU, memory, and I/O devices. Each VM is assigned a virtual machine monitor that has a share of the CPU, memory, and I/O devices to successfully run the VM.

Module 2: Summary

Key points covered in this module:

- Key data center elements
- Application and compute virtualization
- Disk drive components and performance
- Enterprise flash drives
- Host access to storage

This module covered the key elements of a data center – application, DBMS, compute, network, and storage.

It also covered virtualization at application and compute that enable better utilization of resources and ease of management. This module also elaborated on disk drive components and factors governing disk drive performance. It also covered enterprise flash drives that are superior to mechanical disk drives in many ways. This module also covered various options of host to storage access with focus on DAS.

Check Your Knowledge – 1

- Which is a benefit of compute virtualization?
 - A. Enables compute memory swapping
 - B. Improves compute utilization
 - C. Isolates compute memory from the applications
 - D. Isolates compute OS from the applications
- What best describes virtual machines (VMs)?
 - A. All VMs on a physical server must run same OS
 - B. VM files are deleted when VM is powered off
 - C. VMs are discrete sets of files
 - D. All VMs share available resources equally

Check Your Knowledge – 2

- What is concatenation?
 - A. Grouping multiple physical drives into a logical drive
 - B. Dividing a physical drive into multiple logical drives
 - C. Process of writing disk metadata on a logical drive
 - D. Adding more capacity to a physical drive through de-fragmentation

- Which factors contribute to the overall service time of a mechanical disk?
 - A. Disk buffer time, full stroke, and rotation latency
 - B. Internal transfer rate, external transfer rate, and buffer time
 - C. Full stroke, average seek time, and track-to-track seek time
 - D. Average seek time, rotational latency, and data transfer rate

Check Your Knowledge – 3

- Which is a challenge of DAS environment?
 - A. Low performance
 - B. Limited scalability
 - C. Deployment complexity
 - D. High initial investment

Exercise: Design Storage Solution for New Application

- Scenario

- ▶ Characteristics of new application:
 - ▶▶ Require 1TB of storage capacity
 - ▶▶ Peak I/O workload 4900 IOPS
 - ▶▶ Typical I/O size is 4KB
- ▶ Specifications of the available disk drives:
 - ▶▶ 15K rpm drive with storage capacity = 100 GB
 - ▶▶ Average seek time = 5ms
 - ▶▶ Data transfer rate = 40 MB/sec
- ▶ As it is business critical application, response time must be within acceptable range

- Task

- ▶ Calculate the number of disks required for the application

EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 78

An organization is deploying a new business application in their environment. The new application requires 1TB of storage space for business and application data. During peak workload, application is expected to generate 4900 IOPS (I/O per second) with typical I/O size of 4KB.

The available disk drive option is 15,000 rpm drive with 100 GB capacity. Other specification of the drives are:

Av. Seek time = 5 millisecond and data transfer rate = 40MB/sec.

You are asked to calculate the required number of disk drives that can meet both capacity and performance requirements of an application.

Module – 3

Data Protection – RAID



Module 3: Data Protection – RAID

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

- Describe RAID implementation methods
- Describe the three RAID techniques
- Describe commonly used RAID levels
- Describe the impact of RAID on performance
- Compare RAID levels based on their cost, performance, and protection

This module focuses on RAID and its use to improve performance and protection. It details on various RAID implementations, techniques, and levels commonly used. This module also describes the impact of RAID on performance and compares the commonly used RAID levels.

Module 3: Data Protection – RAID

Lesson 1: RAID Overview

During this lesson the following topics are covered:

- RAID Implementation methods
- RAID array components
- RAID techniques

This lesson focuses on RAID implementation methods and RAID array components. This lesson also focuses on various RAID techniques.

Why RAID?

RAID

It is a technique that combines multiple disk drives into a logical unit (RAID set) and provides data protection due to disk failure.

- Due to mechanical components in a disk drive it offers limited performance
- An individual drive has a certain life expectancy and is measured in MTBF:
 - ▶ For example: If the MTBF of a drive is 750,000 hours, and there are 1000 drives in the array, then the MTBF of the array is 750 hours (750,000/1000)
- RAID was introduced to mitigate these problems

Today's data centers house hundreds of disk drives in their storage infrastructure. Disk drives are inherently susceptible to failures due to mechanical wear and tear and other environmental factors, which could result in data loss. The greater the number of disk drives in a storage array, the greater the probability of a disk failure in the array. For example, consider a storage array of 100 disk drives, each with an average life expectancy of 750,000 hours. The average life expectancy of this collection in the array, therefore, is 750,000/100 or 7,500 hours. This means that a disk drive in this array is likely to fail at least once in 7,500 hours.

RAID is an enabling technology that leverages multiple drives as part of a set that provides data protection against drive failures. In general, RAID implementations also improve the storage system performance by serving I/Os from multiple disks simultaneously. Modern arrays with flash drives also benefit in terms of protection and performance by using RAID.

In 1987, Patterson, Gibson, and Katz at the University of California, Berkeley, published a paper titled "A Case for Redundant Arrays of Inexpensive Disks (RAID)." This paper described the use of small-capacity, inexpensive disk drives as an alternative to large-capacity drives common on mainframe computers. The term *RAID* has been redefined to refer to *independent* disks to reflect advances in the storage technology. RAID technology has now grown from an academic concept to an industry standard and is common implementation in today's storage arrays.

RAID Implementation Methods

- Software RAID implementation
 - ▶ Uses host-based software to provide RAID functionality
 - ▶ Limitations
 - ▶▶ Use host CPU cycles to perform RAID calculations, hence impact overall system performance
 - ▶▶ Support limited RAID levels
 - ▶▶ RAID software and OS can be upgraded only if they are compatible
- Hardware RAID Implementation
 - ▶ Uses a specialized hardware controller installed either on a host or on an array

There are two methods of RAID implementation, hardware and software. Both have their advantages and disadvantages.

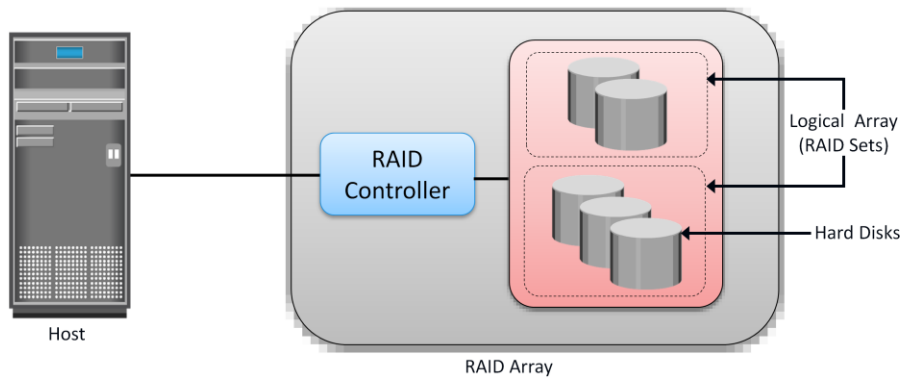
Software RAID uses host-based software to provide RAID functions and is implemented at the operating-system level. Software RAID implementations offer cost and simplicity benefits when compared with hardware RAID. However, they have the following limitations:

- Performance: Software RAID affects overall system performance. This is due to additional CPU cycles required to perform RAID calculations.
- Supported features: Software RAID does not support all RAID levels.
- Operating system compatibility: Software RAID is tied to the host operating system; hence, upgrades to software RAID or to the operating system should be validated for compatibility. This leads to inflexibility in the data-processing environment.

In hardware RAID implementations, a specialized hardware controller is implemented either on the host or on the array. Controller card RAID is a host-based hardware RAID implementation in which a specialized RAID controller is installed in the host, and disk drives are connected to it. Manufacturers also integrate RAID controllers on motherboards. A host-based RAID controller is not an efficient solution in a data center environment with a large number of hosts. The external RAID controller is an array-based hardware RAID. It acts as an interface between the host and disks. It presents storage volumes to the host, and the host manages these volumes as physical drives. The key functions of the RAID controllers are as follows:

- Management and control of disk aggregations
- Translation of I/O requests between logical disks and physical disks
- Data regeneration in the event of disk failures

RAID Array Components



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 84

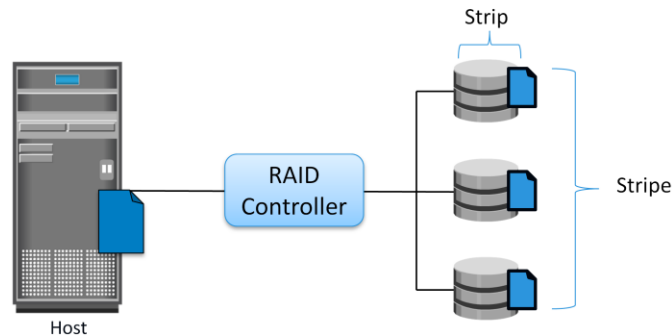
A *RAID array* is an enclosure that contains a number of disk drives and supporting hardware to implement RAID. A subset of disks within a RAID array can be grouped to form logical associations called logical arrays, also known as a *RAID set* or a *RAID group*.

RAID Techniques

- Three key techniques used for RAID are:
 - ▶ Striping
 - ▶ Mirroring
 - ▶ Parity

RAID techniques – striping, mirroring, and parity – form the basis for defining various RAID levels. These techniques determine the data availability and performance characteristics of a RAID set.

RAID Technique – Striping



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 86

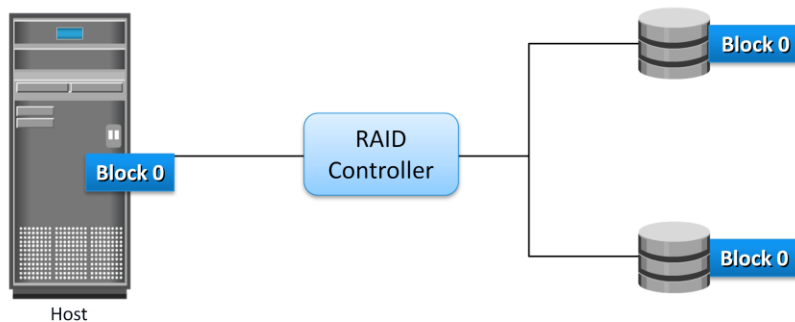
Striping is a technique of spreading data across multiple drives (more than one) in order to use the drives in parallel. All the read-write heads work simultaneously, allowing more data to be processed in a shorter time and increasing performance, compared to reading and writing from a single disk.

Within each disk in a RAID set, a predefined number of contiguously addressable disk blocks are defined as strip. The set of aligned strips that spans across all the disks within the RAID set is called a stripe. Figure on the slide shows physical and logical representations of a striped RAID set.

Strip size (also called *stripe depth*) describes the number of blocks in a *strip*, and is the maximum amount of data that can be written to or read from a single disk in the set, assuming that the accessed data starts at the beginning of the strip. All strips in a stripe have the same number of blocks. Having a smaller strip size means that the data is broken into smaller pieces while spread across the disks.

Stripe size is a multiple of strip size by the number of data disks in the RAID set. For example, in a five disk striped RAID set with a strip size of 64KB, the stripe size is 320 KB (64KB x 5). *Stripe width* refers to the number of data strips in a stripe. Striped RAID does not provide any data protection unless parity or mirroring is used.

RAID Technique – Mirroring



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 87

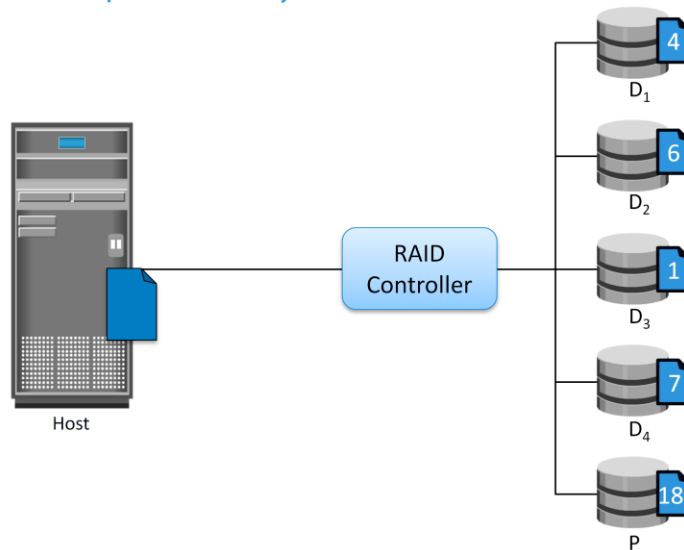
Mirroring is a technique whereby the same data is stored on two different disk drives, yielding two copies of the data. If one disk drive failure occurs, the data is intact on the surviving disk drive and the controller continues to service the host's data requests from the surviving disk of a mirrored pair.

When the failed disk is replaced with a new disk, the controller copies the data from the surviving disk of the mirrored pair. This activity is transparent to the host.

In addition to providing complete data redundancy, mirroring enables fast recovery from disk failure. However, disk mirroring provides only data protection and is not a substitute for data backup. Mirroring constantly captures changes in the data, whereas a backup captures point-in-time images of the data.

Mirroring involves duplication of data—the amount of storage capacity needed is twice the amount of data being stored. Therefore, mirroring is considered expensive and is preferred for mission-critical applications that cannot afford the risk of any data loss. Mirroring improves read performance because read requests can be serviced by both disks. However, write performance is slightly lower than that in a single disk because each write request manifests as two writes on the disk drives. Mirroring does not deliver the same levels of write performance as a striped RAID.

RAID Technique – Parity



Actual parity calculation is a bitwise XOR operation

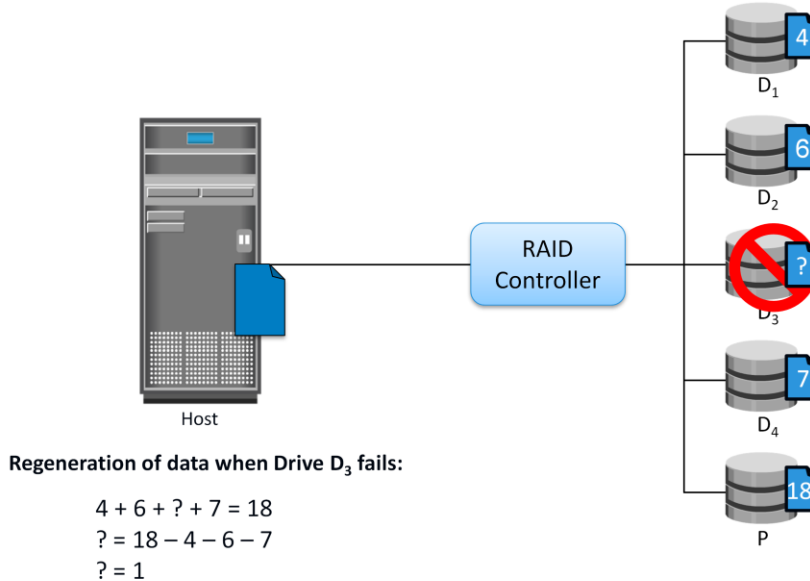
EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 88

Parity is a method to protect striped data from disk drive failure without the cost of mirroring. An additional disk drive is added to hold parity, a mathematical construct that allows re-creation of the missing data. Parity is a redundancy technique that ensures protection of data without maintaining a full set of duplicate data. Calculation of parity is a function of the RAID controller.

Parity information can be stored on separate, dedicated disk drives or distributed across all the drives in a RAID set. The first four disks in the figure, labeled D_1 to D_4 , contain the data. The fifth disk, labeled P , stores the parity information, which, in this case, is the sum of the elements in each row.

Data Recovery in Parity Technique



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 89

Now, if one of the data disks fails, the missing value can be calculated by subtracting the sum of the rest of the elements from the parity value. Here, for simplicity, the computation of parity is represented as an arithmetic sum of the data. However, parity calculation is a *bitwise XOR* operation.

Compared to mirroring, parity implementation considerably reduces the cost associated with data protection. Consider an example of a parity RAID configuration with five disks where four disks hold data, and the fifth holds the parity information. In this example, parity requires only 25 percent extra disk space compared to mirroring, which requires 100 percent extra disk space. However, there are some disadvantages of using parity. Parity information is generated from data on the data disk. Therefore, parity is recalculated every time there is a change in data. This recalculation is time-consuming and affects the performance of the RAID array.

For parity RAID, the stripe size calculation does not include the parity strip. For example in a five (4 + 1) disk parity RAID set with a strip size of 64 KB, the stripe size will be 256 KB (64 KB x 4).

Module 3: Data Protection – RAID

Lesson 2: RAID Levels

During this lesson the following topics are covered:

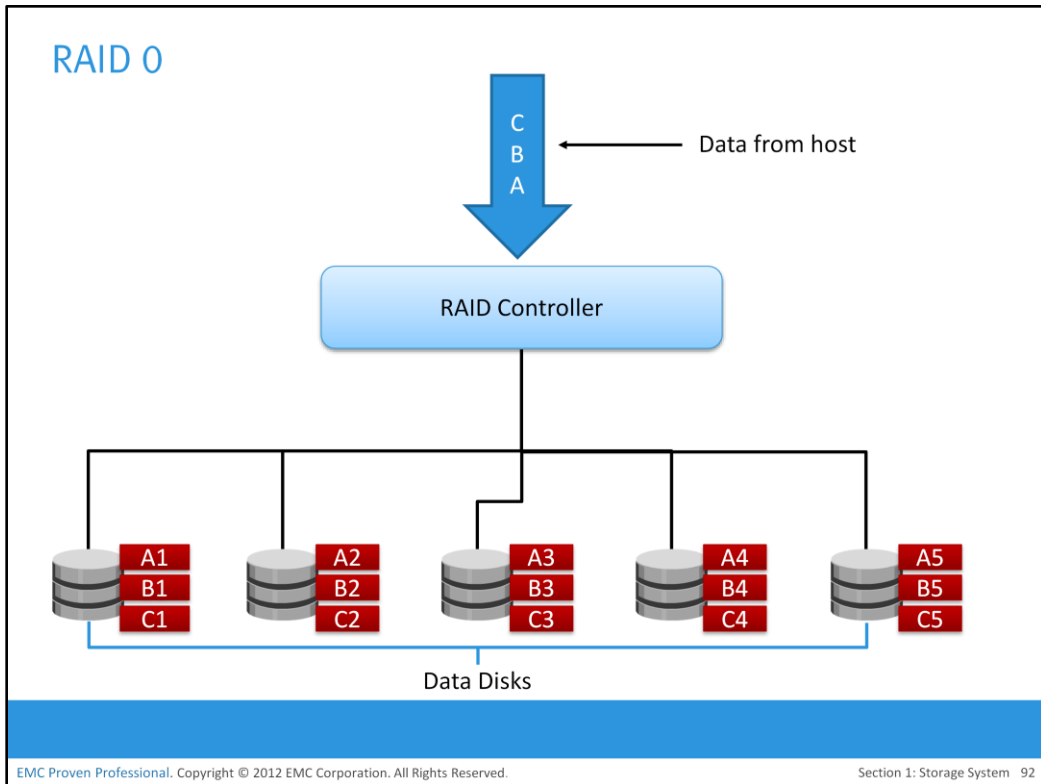
- Commonly used RAID levels
- RAID impacts on performance
- RAID comparison
- Hot spare

This lesson focuses on commonly used RAID levels and their comparisons. This lesson also focuses on Hot spare.

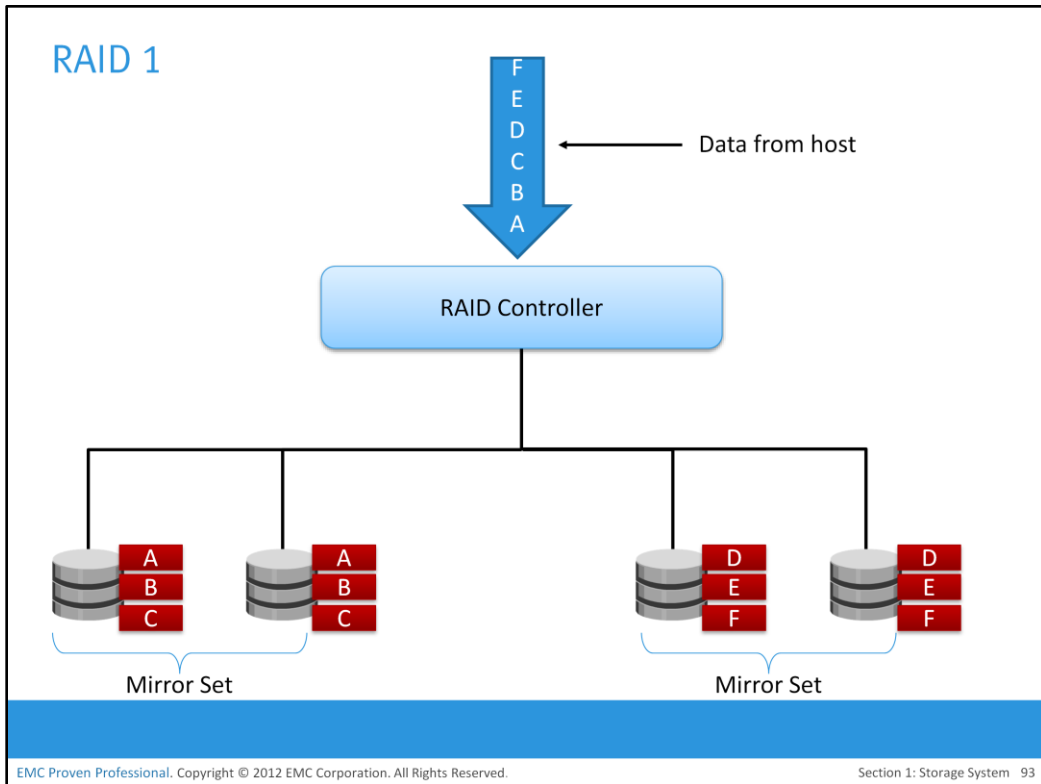
RAID Levels

- Commonly used RAID levels are:
 - ▶ RAID 0 – Striped set with no fault tolerance
 - ▶ RAID 1 – Disk mirroring
 - ▶ RAID 1 + 0 – Nested RAID
 - ▶ RAID 3 – Striped set with parallel access and dedicated parity disk
 - ▶ RAID 5 – Striped set with independent disk access and a distributed parity
 - ▶ RAID 6 – Striped set with independent disk access and dual distributed parity

Application performance, data availability requirements, and cost determine the RAID level selection. These RAID levels are defined on the basis of striping, mirroring, and parity techniques. Some RAID levels use a single technique, whereas others use a combination of techniques. The commonly used RAID levels are listed on the slide.

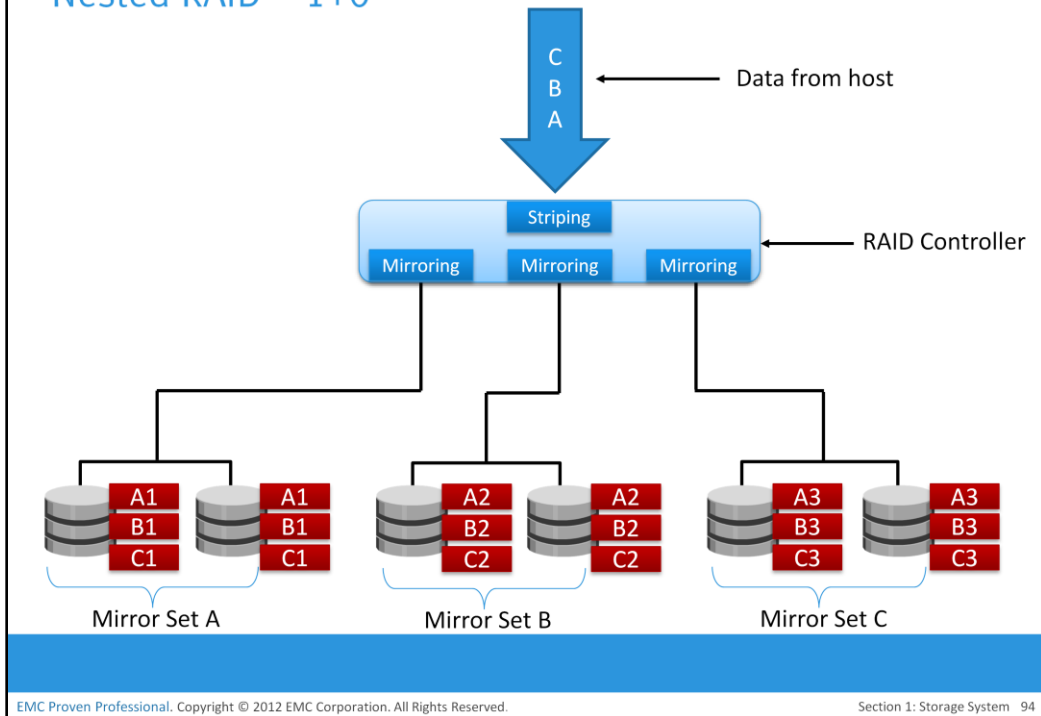


RAID 0 configuration uses data striping techniques, where data is striped across all the disks within a RAID set. Therefore, it utilizes the full storage capacity of a RAID set. To read data, all the strips are put back together by the controller. When the number of drives in the RAID set increases, performance improves because more data can be read or written simultaneously. RAID 0 is a good option for applications that need high I/O throughput. However, if these applications require high availability during drive failures, RAID 0 does not provide data protection and availability.



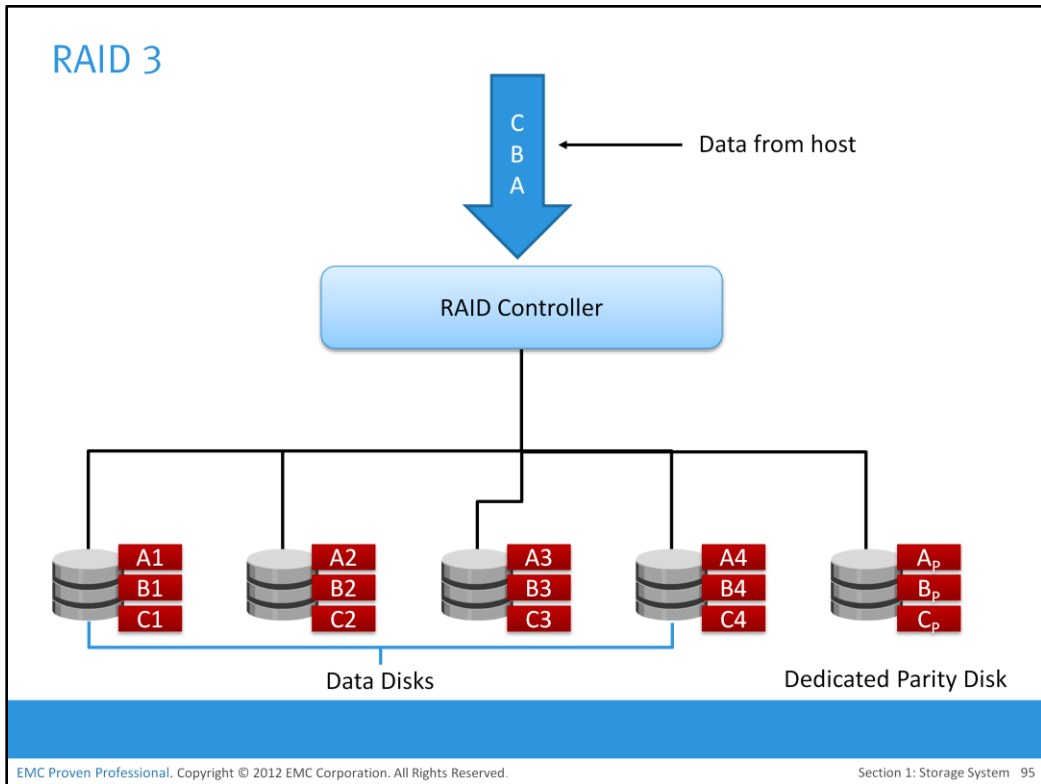
RAID 1 is based on the mirroring technique. In this RAID configuration, data is mirrored to provide fault tolerance. A RAID 1 set consists of two disk drives and every write is written to both disks. The mirroring is transparent to the host. During disk failure, the impact on data recovery in RAID 1 is the least among all RAID implementations. This is because the RAID controller uses the mirror drive for data recovery. RAID 1 is suitable for applications that require high availability and cost is no constraint.

Nested RAID – 1+0



Most data centers require data redundancy and performance from their RAID arrays. RAID 1+0 combines the performance benefits of RAID 0 with the redundancy benefits of RAID 1. It uses mirroring and striping techniques and combine their benefits. This RAID type requires an even number of disks, the minimum being four.

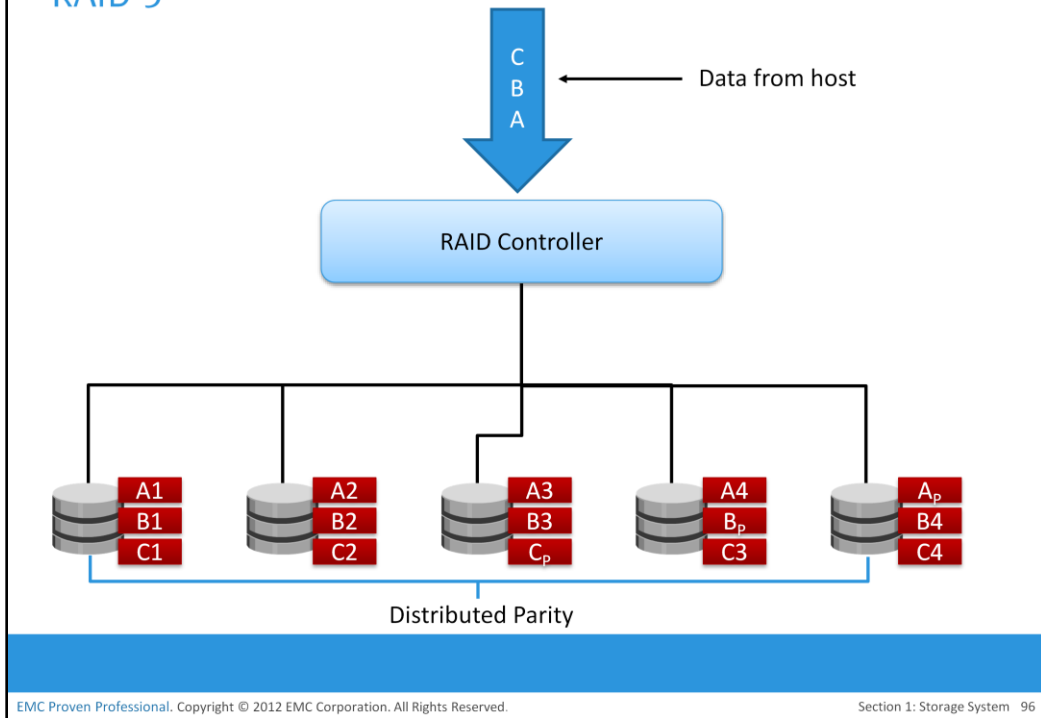
RAID 1+0 is also known as RAID 10 (Ten) or RAID 1/0. RAID 1+0 is also called striped mirror. The basic element of RAID 1+0 is a mirrored pair, which means that data is first mirrored and then both copies of the data are striped across multiple disk drive pairs in a RAID set. When replacing a failed drive, only the mirror is rebuilt. In other words, the disk array controller uses the surviving drive in the mirrored pair for data recovery and continuous operation. Data from the surviving disk is copied to the replacement disk.



RAID 3 stripes data for performance and uses parity for fault tolerance. Parity information is stored on a dedicated drive so that the data can be reconstructed if a drive fails in a RAID set. For example, in a set of five disks, four are used for data and one for parity. Therefore, the total disk space required is 1.25 times the size of the data disks. RAID 3 always reads and writes complete stripes of data across all disks because the drives operate in parallel. There are no partial writes that update one out of many strips in a stripe.

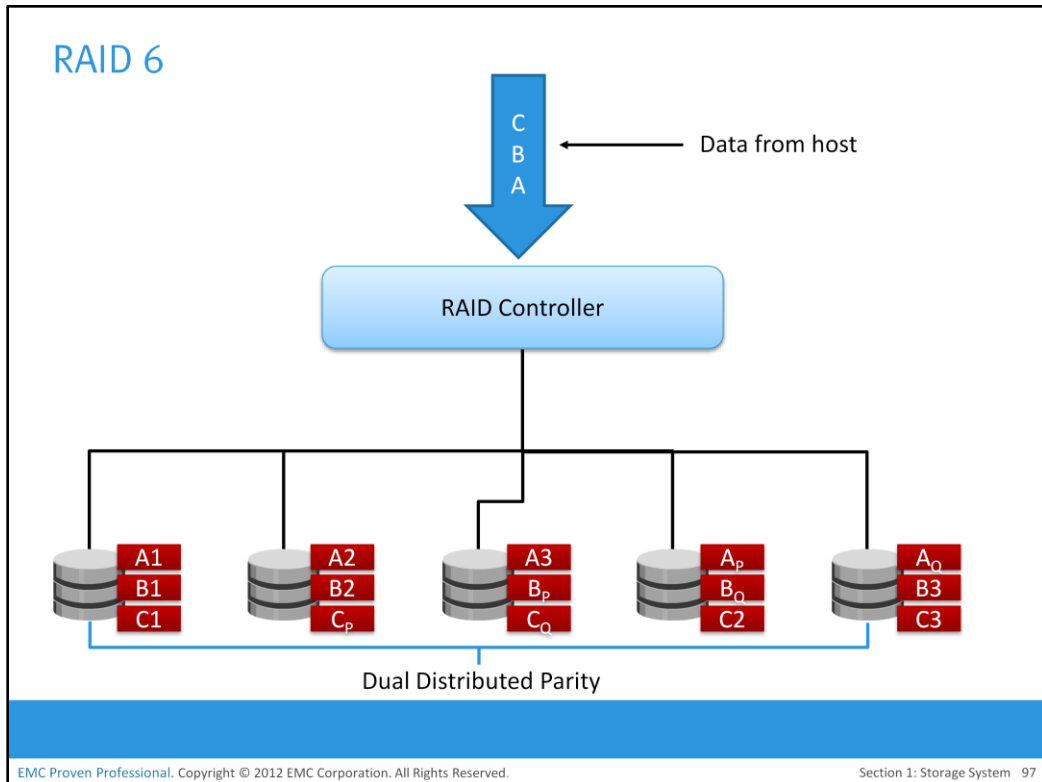
Similar to RAID 3, RAID 4 stripes data for high performance and uses parity for improved fault tolerance. Data is striped across all disks except the parity disk in the array. Parity information is stored on a dedicated disk so that the data can be rebuilt if a drive fails. Unlike RAID 3, data disks in RAID 4 can be accessed independently so that specific data elements can be read or written on a single disk without reading or writing an entire stripe. RAID 4 provides good read throughput and reasonable write throughput.

RAID 5



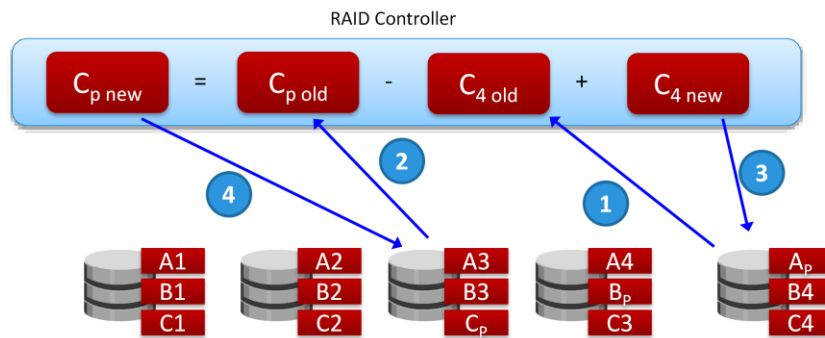
RAID 5 is a versatile RAID implementation. It is similar to RAID 4 because it uses striping. The drives (strips) are also independently accessible. The difference between RAID 4 and RAID 5 is the parity location. In RAID 4, parity is written to a dedicated drive, creating a write bottleneck for the parity disk. In RAID 5, parity is distributed across all disks to overcome the write bottleneck of a dedicated parity disk.

RAID 6



RAID 6 works the same way as RAID 5, except that RAID 6 includes a second parity element to enable survival if two disk failures occur in a RAID set. Therefore, a RAID 6 implementation requires at least four disks. RAID 6 distributes the parity across all the disks. The write penalty (explained later in this module) in RAID 6 is more than that in RAID 5; therefore, RAID 5 writes perform better than RAID 6. The rebuild operation in RAID 6 may take longer than that in RAID 5 due to the presence of two parity sets.

RAID Impacts on Performance



- In RAID 5, every write (update) to a disk manifests as four I/O operations (two disk reads and two disk writes)
- In RAID 6, every write (update) to a disk manifests as six I/O operations (three disk reads and three disk writes)
- In RAID 1, every write manifests as two I/O operations (two disk writes)

When choosing a RAID type, it is imperative to consider its impact on disk performance and application IOPS. In both mirrored and parity RAID configurations, every write operation translates into more I/O overhead for the disks, which is referred to as a *write penalty*. In a RAID 1 implementation, every write operation must be performed on two disks configured as a mirrored pair, whereas in a RAID 5 implementation, a write operation may manifest as four I/O operations. When performing I/Os to a disk configured with RAID 5, the controller has to read, recalculate, and write a parity segment for every data write operation.

This slide illustrates a single write operation on RAID 5 that contains a group of five disks. The parity (P) at the controller is calculated as follows:

$$C_p = C_1 + C_2 + C_3 + C_4 \text{ (XOR operations)}$$

Whenever the controller performs a write I/O, parity must be computed by reading the old parity (C_p old) and the old data ($C_{4 \text{ old}}$) from the disk, which means two read I/Os. Then, the new parity (C_p new) is computed as follows:

$$C_{p \text{ new}} = C_{p \text{ old}} - C_{4 \text{ old}} + C_{4 \text{ new}} \text{ (XOR operations)}$$

After computing the new parity, the controller completes the write I/O by writing the new data and the new parity onto the disks, amounting to two write I/Os. Therefore, the controller performs two disk reads and two disk writes for every write operation, and the write penalty is 4.

In RAID 6, which maintains dual parity, a disk write requires three read operations: two parity and one data. After calculating both new parities, the controller performs three write operations: two parity and an I/O. Therefore, in a RAID 6 implementation, the controller performs six I/O operations for each write I/O, and the write penalty is 6.

RAID Penalty Calculation Example

- Total IOPS at peak workload is 1200
- Read/Write ratio 2:1
- Calculate disk load at peak activity for:
 - ▶ RAID 1/0
 - ▶ RAID 5

Consider an application that generates 1200 IOPS at peak workload, with read/write ratio of 2:1. Calculate disk load at peak activity for RAID 1/0 and RAID 5 configuration.

Solution: RAID Penalty

- For RAID 1/0, the disk load (read + write)
$$= (1200 \times 2/3) + (1200 \times (1/3) \times 2)$$
$$= 800 + 800$$
$$= 1600 \text{ IOPS}$$
- For RAID 5, the disk load (read + write)
$$= (1200 \times 2/3) + (1200 \times (1/3) \times 4)$$
$$= 800 + 1600$$
$$= 2400 \text{ IOPS}$$

The read-write ratio for an application is given as 2:1, and the total I/O generated by the application is 1200. Therefore, in case of RAID 1/0 the total reads are 800 and writes are 400. However, for RAID 1/0 write penalty is 2. Therefore, the total disk load for writes will be 800. Hence, the total disk load will be 1600.

Similarly, the disk load for RAID 5 can be calculated. The write penalty for RAID 5 is 4, therefore the total IOPS will be 800 reads and 1600 writes, and the total disk load will be 2400 IOPS.

RAID Comparison

RAID level	Min disks	Available storage capacity (%)	Read performance	Write performance	Write penalty	Protection
1	2	50	Better than single disk	Slower than single disk, because every write must be committed to all disks	Moderate	Mirror
1+0	4	50	Good	Good	Moderate	Mirror
3	3	$[(n-1)/n]*100$	Fair for random reads and good for sequential reads	Poor to fair for small random writes fair for large, sequential writes	High	Parity (Supports single disk failure)
5	3	$[(n-1)/n]*100$	Good for random and sequential reads	Fair for random and sequential writes	High	Parity (Supports single disk failure)
6	4	$[(n-2)/n]*100$	Good for random and sequential reads	Poor to fair for random and sequential writes	Very High	Parity (Supports two disk failures)

where n = number of disks

The table on the slide compare different RAID levels.

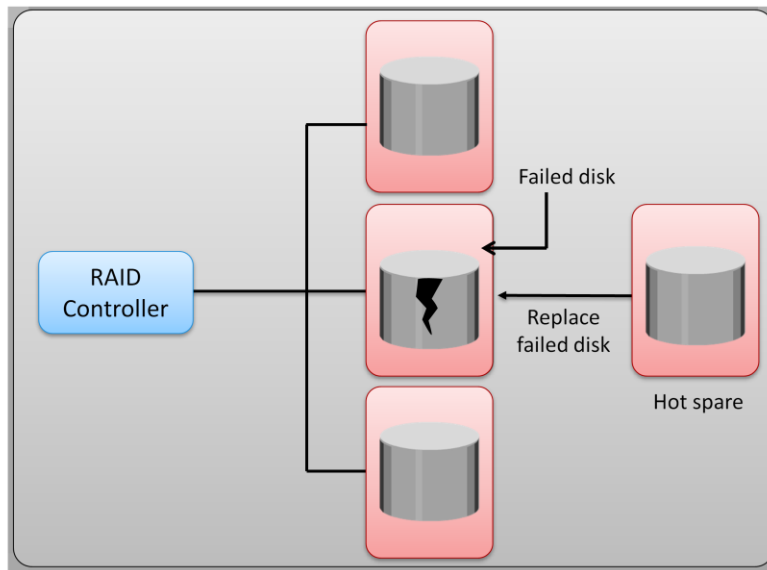
Suitable RAID Levels for Different Applications

- RAID 1+0
 - ▶ Suitable for applications with small, random, and write intensive (writes typically greater than 30%) I/O profile
 - ▶ Example: OLTP, RDBMS – Temp space
- RAID 3
 - ▶ Large, sequential read and write
 - ▶ Example: data backup and multimedia streaming
- RAID 5 and 6
 - ▶ Small, random workload (writes typically less than 30%)
 - ▶ Example: email, RDBMS – Data entry

Common applications that benefit from different RAID levels.

- RAID 1+0 performs well for workloads that use small, random, write-intensive I/Os. Some applications that benefit from RAID 1+0 are high transaction rate online transaction processing (OLTP), RDBMS temp space and so on.
- RAID 3 provides good performance for applications that involve large sequential data access, such as data backup or video streaming.
- RAID 5 is good for random, read intensive I/O applications and preferred for messaging, medium-performance media serving, and relational database management system (RDBMS) implementations, in which database administrators (DBAs) optimize data access.

Hot Spare



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 103

A *hot spare* refers to a spare drive in a RAID array that temporarily replaces a failed disk drive by taking the identity of the failed disk drive. With the hot spare, one of the following methods of data recovery is performed depending on the RAID implementation:

- If parity RAID is used, the data is rebuilt onto the hot spare from the parity and the data on the surviving disk drives in the RAID set.
- If mirroring is used, the data from the surviving mirror is used to copy the data onto the hot spare.

When a new disk drive is added to the system, data from the hot spare is copied to it. The hot spare returns to its idle state, ready to replace the next failed drive. Alternatively, the hot spare replaces the failed disk drive permanently. This means that it is no longer a hot spare, and a new hot spare must be configured on the array.

A hot spare should be large enough to accommodate data from a failed drive. Some systems implement multiple hot spares to improve data availability.

A hot spare can be configured as automatic or user initiated, which specifies how it will be used in the event of disk failure. In an automatic configuration, when the recoverable error rates for a disk exceed a predetermined threshold, the disk subsystem tries to copy data from the failing disk to the hot spare automatically. If this task is completed before the damaged disk fails, the subsystem switches to the hot spare and marks the failing disk as unusable. Otherwise, it uses parity or the mirrored disk to recover the data. In the case of a user-initiated configuration, the administrator has control of the rebuild process. For example, the rebuild could occur overnight to prevent any degradation of system performance. However, the system is at risk of data loss if another disk failure occurs.

Module 3: Summary

Key points covered in this module:

- RAID implementation methods and techniques
- Common RAID levels
- RAID write penalty
- Compare RAID levels based on their cost and performance

This module covered the two methods of RAID implementation, hardware and software. The three techniques on which the RAID levels are built are striping, mirroring, and parity. The commonly used RAID levels are 0, 1, 1+0, 3, 5, and 6.

When choosing a RAID type, it is imperative to consider its impact on disk performance and application IOPS. In both mirrored and parity RAID configurations, every write operation translates into more I/O overhead for the disks, which is referred to as a *write penalty*.

Finally, this module compared different RAID levels based on their cost, performance, and write penalty.

Check Your Knowledge – 1

- Which statement is true about software RAID implementation?
 - A. Upgrades to operating system do not require compatibility validation with RAID software
 - B. It is expensive than hardware RAID implementation
 - C. Supports all RAID levels
 - D. Uses host CPU cycles to perform RAID calculations
- An application generates 400 small random IOPS with a read/write ratio of 3:1. What is the RAID-corrected IOPS on the disk for RAID 5 ?
 - A. 400
 - B. 500
 - C. 700
 - D. 900

Check Your Knowledge – 2

- What is write penalty in a RAID 6 configuration for small random I/Os?
 - A. 2
 - B. 3
 - C. 4
 - D. 6
- Which application is most benefited by using RAID 3?
 - A. Backup
 - B. OLTP
 - C. e-commerce
 - D. email

Check Your Knowledge – 3

- What is the stripe size of a five disk parity RAID 5 set that has a strip size of 64 KB?
 - A. 64 KB
 - B. 128 KB
 - C. 256 KB
 - D. 320 KB

Exercise 1: RAID

- A company is planning to reconfigure storage for their accounting application for high availability
 - ▶ Current configuration and challenges
 - ▶▶ Application performs 15% random writes and 85% random reads
 - ▶▶ Currently deployed with five disk RAID 0 configuration
 - ▶▶ Each disk has an advertised formatted capacity of 200 GB
 - ▶▶ Total size of accounting application's data is 730 GB which is unlikely to change over 6 months
 - ▶▶ Approaching end of financial year, buying even one disk is not possible
 - Task
 - ▶ Recommend a RAID level that the company can use to restructure their environment fulfilling their needs
 - ▶ Justify your choice based on cost, performance, and availability

EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 108

Business Profile:

A company, involved in mobile wireless services across the country, has about 5000 employees worldwide. This company has 7 regional offices across the country. Although the company is financially doing well, they continue to feel the competitive pressure. As a result, the company needs to ensure that the IT infrastructure takes advantage of fault tolerant features.

Current Configuration and Challenges:

The company uses different applications for communication, accounting, and management. All the applications are hosted on individual servers with disks configured as RAID 0. All financial activity is managed and tracked by a single accounting application. It is very important for the accounting data to be highly available. The application performs around 15% random write operations and the remaining 85% are random reads. The accounting data is currently stored on a 5-disk RAID 0 set. Each disk has an advertised formatted capacity of 200 GB and the total size of their files is 730 GB. The company performs nightly backups and removes old information — so the amount of data is unlikely to change much over the next 6 months. The company is approaching the end of the financial year and the IT budget is depleted. It won't be possible to buy even one new disk drive.

Tasks:

Recommend a RAID level that the company can use to restructure their environment fulfilling their needs.

Justify your choice based on cost, performance, and availability of the new solution.

Exercise 2: RAID

- A company (same as discussed in exercise 1) is now planning to reconfigure storage for their database application for HA
 - ▶ Current configuration and challenges
 - ▶▶ The application performs 40% writes and 60% reads
 - ▶▶ Currently deployed on six disk RAID 0 configuration with advertised capacity of each disk being 200 GB
 - ▶▶ Size of the database is 900 GB and amount of data is likely to change by 30% over the next 6 months
 - ▶▶ It is a new financial year and the company has an increased budget
 - Task
 - ▶ Recommend a suitable RAID level to fulfill company's needs
 - ▶ Estimate the cost of the new solution (200GB disk costs \$1000)
 - ▶ Justify your choice based on cost, performance, and availability

Business Profile:

A company, involved in mobile wireless services across the country, has about 5000 employees worldwide. This company has 7 regional offices across the country. Although the company is financially doing well, they continue to feel the competitive pressure. As a result, the company needs to ensure that the IT infrastructure takes advantage of fault tolerant features.

Current Configuration and Challenges:

The company uses an accounting application that is hosted on an individual server with disks configured as RAID 0. It is now the beginning of a new financial year and the IT department has an increased budget. You are called in to recommend changes to their database environment. You investigate their database environment closely and observe that the data is stored on a 6-disk RAID 0 set. Each disk has an advertised formatted capacity of 200 GB and the total size of their files is 900 GB. The amount of data is likely to change by 30 % over the next 6 months and your solution must accommodate this growth. The application performs around 40% write operations and the remaining 60 % are reads.

Tasks:

Recommend a RAID level that the company can use to restructure their environment and fulfill their needs. What is the cost of the new solution?

Justify your choice based on cost, performance, and data availability of the new solution.

Note: A new 200 GB disk drive costs \$1000. The controller can handle all commonly used RAID levels, so will not need to be replaced.

Module – 4

Intelligent Storage System



Module 4: Intelligent Storage System

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

- Describe the key components of intelligent storage system
- Describe cache management and protection techniques
- Describe two storage provisioning methods
- Describe two types of intelligent storage systems

This module focuses on the key components of an intelligent storage system. It details the function of each component, including cache management and protection techniques. The module also focuses on the two storage provisioning methods. Finally, it describes the two types of intelligent storage systems.

Module 4: Intelligent Storage System

Lesson 1: Key Components of an Intelligent Storage System

During this lesson the following topics are covered:

- Intelligent storage system overview
- Key components of an intelligent storage system
- Cache management

This lesson focuses on intelligent storage system overview and key components of an intelligent storage system. This lesson also focuses on cache management.

What is an Intelligent Storage System (ISS) ?

Intelligent Storage System

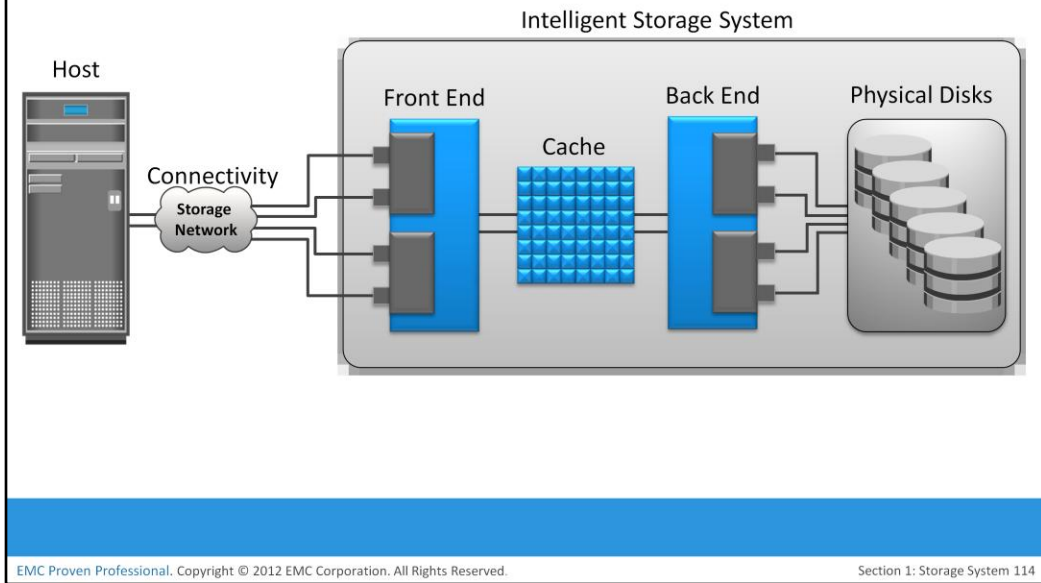
It is a feature-rich RAID array that provides highly optimized I/O processing capabilities.

- Provides large amount of cache and multiple I/O paths that enhances the performance
- Has an operating environment that provides
 - ▶ Intelligent cache management
 - ▶ Array resource management
 - ▶ Connectivity to heterogeneous hosts
- Supports flash drive, virtual provisioning, and automated storage tiering

Business-critical applications require high levels of performance, availability, security, and scalability. A disk drive is a core element of storage that governs the performance of any storage system. Some of the older disk-array technologies could not overcome performance constraints due to the limitations of disk drives and their mechanical components. RAID technology made an important contribution to enhancing storage performance and reliability, but disk drives, even with a RAID implementation, could not meet the performance requirements of today's applications.

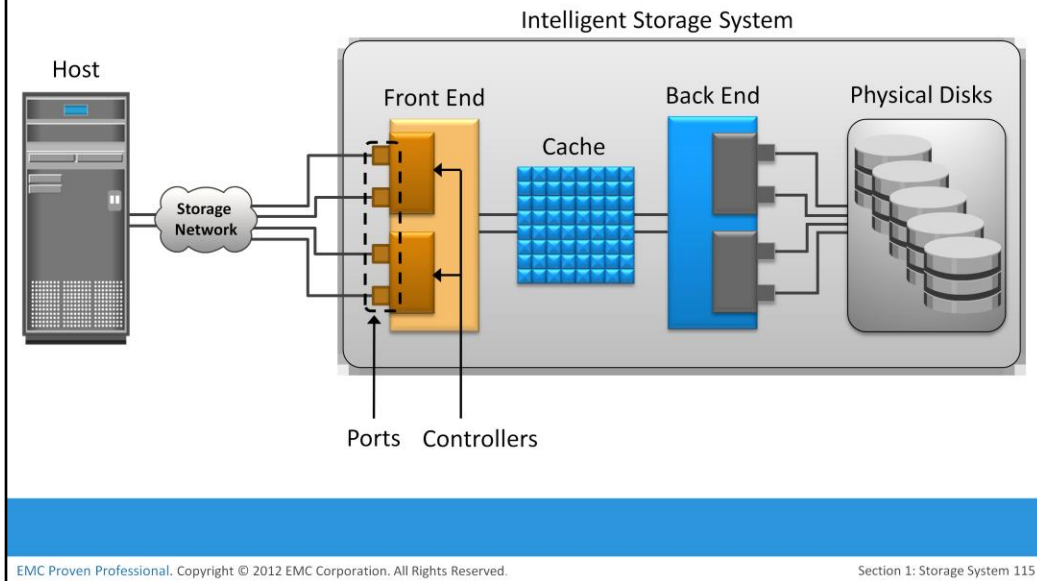
With advancements in technology, a new breed of storage solutions, known as *intelligent storage systems*, has evolved. These intelligent storage systems are feature-rich RAID arrays that provide highly optimized I/O processing capabilities. These storage systems are configured with a large amount of memory (called *cache*) and multiple I/O paths and use sophisticated algorithms to meet the requirements of performance-sensitive applications. These arrays have an operating environment that intelligently and optimally handles the management, allocation, and utilization of storage resources. Support for flash drives and other modern-day technologies, such as virtual storage provisioning and automated storage tiering, has added a new dimension to storage system performance, scalability, and availability.

Key Components of an ISS



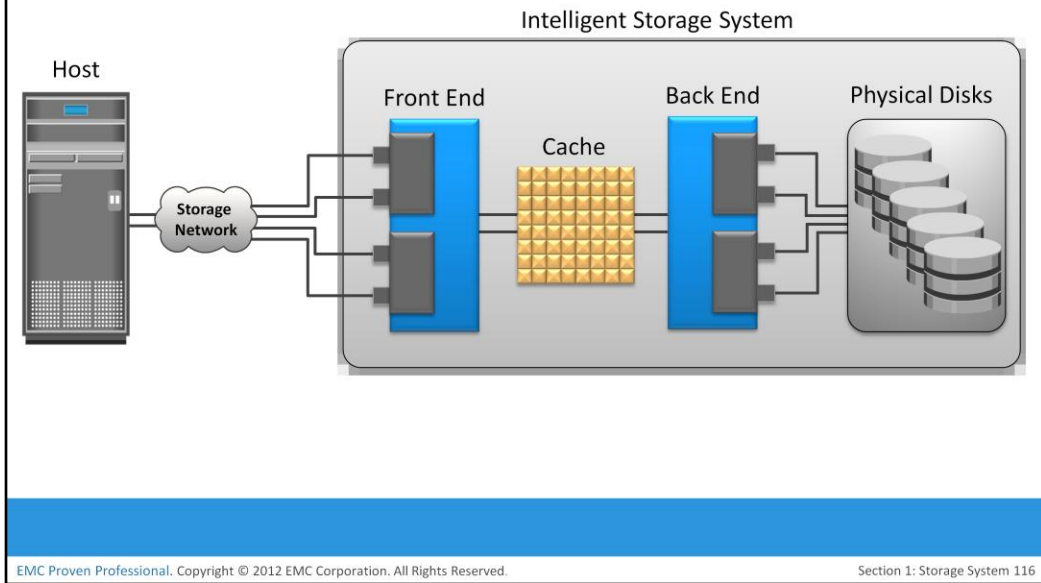
An intelligent storage system consists of four key components: *front end*, *cache*, *back end*, and *physical disks*. An I/O request received from the host at the front-end port is processed through cache and back end, to enable storage and retrieval of data from the physical disk. A read request can be serviced directly from cache if the requested data is found in the cache. In modern intelligent storage systems, front end, cache, and back end are typically integrated on a single board (referred as a *storage processor* or *storage controller*).

Key Components of ISS: Front End



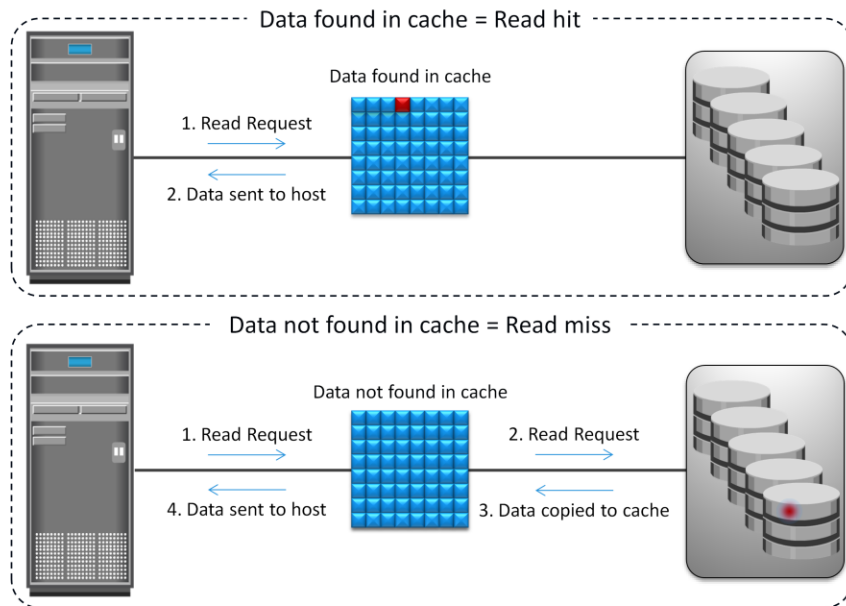
The front end provides the interface between the storage system and the host. It consists of two components: front-end ports and front-end controllers. Typically, a front end has redundant controllers for high availability, and each controller contains multiple ports that enable large numbers of hosts to connect to the intelligent storage system. Each front-end controller has processing logic that executes the appropriate transport protocol, such as Fibre Channel, iSCSI, FICON, or FCoE for storage connections. *Front-end controllers* route data to and from cache via the internal data bus. When the cache receives the write data, the controller sends an acknowledgment message back to the host.

Key Components of ISS: Cache



Cache is semiconductor memory where data is placed temporarily to reduce the time required to service I/O requests from the host. Cache improves storage system performance by isolating hosts from the mechanical delays associated with rotating disks or hard disk drive (HDD). Rotating disks are the slowest component of an intelligent storage system. Data access on rotating disks usually takes several milliseconds because of seek time and rotational latency. Accessing data from cache is fast and typically takes less than a millisecond. On intelligent arrays, write data is first placed in cache and then written to disk.

Read Operation with Cache



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

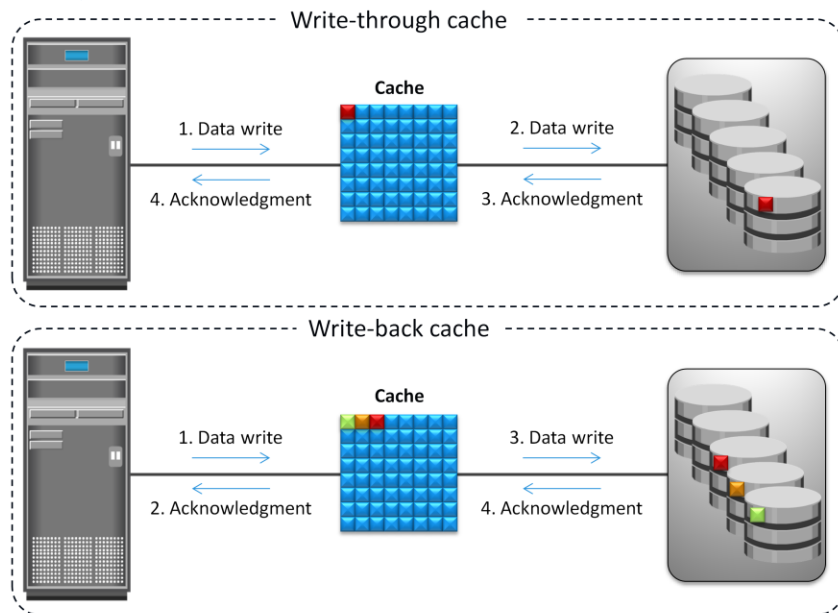
Section 1: Storage System 117

When a host issues a read request, the storage controller reads the tag RAM to determine whether the required data is available in cache. If the requested data is found in the cache, it is called a *read cache hit* or *read hit* and data is sent directly to the host, without any disk operation. This provides a fast response time to the host (about a millisecond). If the requested data is not found in cache, it is called a *cache miss* and the data must be read from the disk. The back end accesses the appropriate disk and retrieves the requested data. Data is then placed in cache and finally sent to the host through the front end. Cache misses increase the I/O response time.

A *prefetch* or *read-ahead* algorithm is used when read requests are sequential. In a sequential read request, a contiguous set of associated blocks is retrieved. Several other blocks that have not yet been requested by the host can be read from the disk and placed into cache in advance. When the host subsequently requests these blocks, the read operations will be read hits. This process significantly improves the response time experienced by the host. The intelligent storage system offers fixed and variable prefetch sizes. In *fixed prefetch*, the intelligent storage system prefetches a fixed amount of data. It is most suitable when host I/O sizes are uniform. In *variable prefetch*, the storage system prefetches an amount of data in multiples of the size of the host request. *Maximum prefetch* limits the number of data blocks that can be prefetched to prevent the disks from being rendered busy with prefetch at the expense of other I/Os.

Read performance is measured in terms of the *read hit ratio* or the *hit rate*, usually expressed as a percentage. This ratio is the number of read hits with respect to the total number of read requests. A higher read hit ratio improves the read performance.

Write Operation with Cache



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 118

Write operations with cache provide performance advantages over writing directly to disks. When an I/O is written to cache and acknowledged, it is completed in far less time (from the host's perspective) than it would take to write directly to disk. Sequential writes also offer opportunities for optimization because many smaller writes can be coalesced for larger transfers to disk drives with the use of cache.

A write operation with cache is implemented in the following ways:

- **Write-back cache:** Data is placed in cache and an acknowledgment is sent to the host immediately. Later, data from several writes are committed (de-staged) to the disk. Write response times are much faster because the write operations are isolated from the mechanical delays of the disk. However, uncommitted data is at risk of loss if cache failures occur.
- **Write-through cache:** Data is placed in the cache and immediately written to the disk, and an acknowledgment is sent to the host. Because data is committed to disk as it arrives, the risks of data loss are low, but the write-response time is longer because of the disk operations.

Cache can be bypassed under certain conditions, such as large size write I/O. In this implementation, if the size of an I/O request exceeds the predefined size, called *write aside size*, writes are sent to the disk directly to reduce the impact of large writes consuming a large cache space. This is particularly useful in an environment where cache resources are constrained and cache is required for small random I/Os.

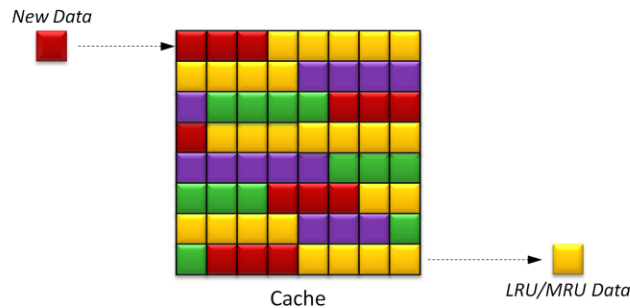
Cont..

Cache can be implemented as either dedicated cache or global cache. With dedicated cache, separate sets of memory locations are reserved for reads and writes. In global cache, both reads and writes can use any of the available memory addresses. Cache management is more efficient in a global cache implementation because only one global set of addresses has to be managed.

Global cache allows users to specify the percentages of cache available for reads and writes for cache management. Typically, the read cache is small, but it should be increased if the application being used is read-intensive. In other global cache implementations, the ratio of cache available for reads versus writes is dynamically adjusted based on the workloads.

Cache Management: Algorithms

- Least recently used (LRU)
 - ▶ Discards data that have not been accessed for a long time
- Most recently used (MRU)
 - ▶ Discards data that have been most recently accessed



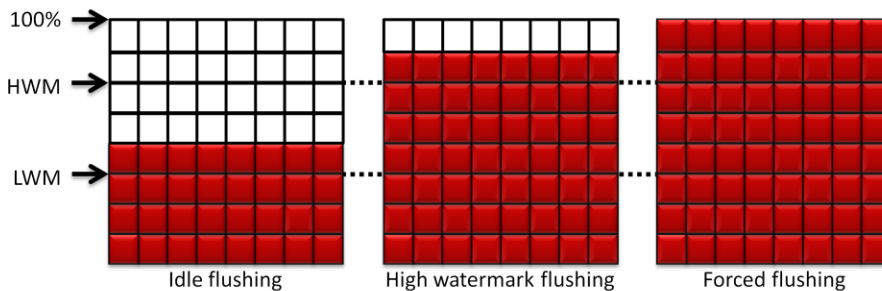
Cache is a finite and expensive resource that needs proper management. Even though modern intelligent storage systems come with a large amount of cache, when all cache pages are filled, some pages have to be freed up to accommodate new data and avoid performance degradation. Various cache management algorithms are implemented in intelligent storage systems to proactively maintain a set of free pages and a list of pages that can be potentially freed up whenever required.

The most commonly used algorithms are discussed in the following list:

- **Least Recently Used (LRU):** An algorithm that continuously monitors data access in cache and identifies the cache pages that have not been accessed for a long time. LRU either frees up these pages or marks them for reuse. This algorithm is based on the assumption that data that has not been accessed for a while will not be requested by the host. However, if a page contains write data that has not yet been committed to disk, the data is first written to disk before the page is reused.
- **Most Recently Used (MRU):** This algorithm is the opposite of LRU, where the pages that have been accessed most recently are freed up or marked for reuse. This algorithm is based on the assumption that recently accessed data may not be required for a while.

Cache Management: Watermarking

- Manages I/O burst through flushing process
 - ▶ Flushing is the process of committing data from cache to the disk
- Three modes of flushing to manage cache utilization are:
 - ▶ Idle flushing
 - ▶ High watermark flushing
 - ▶ Forced flushing



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 121

As cache fills, the storage system must take action to flush dirty pages (data written into the cache but not yet written to the disk) to manage space availability. *Flushing* is the process that commits data from cache to the disk. On the basis of the I/O access rate and pattern, high and low levels called *watermarks* are set in cache to manage the flushing process. *High watermark (HWM)* is the cache utilization level at which the storage system starts high-speed flushing of cache data. *Low watermark (LWM)* is the point at which the storage system stops flushing data to the disks. The cache utilization level drives the mode of flushing to be used:

- Idle flushing: Occurs continuously, at a modest rate, when the cache utilization level is between the high and low watermark.
- High watermark flushing: Activated when cache utilization hits the high watermark. The storage system dedicates some additional resources for flushing. This type of flushing has some impact on I/O processing.
- Forced flushing: Occurs in the event of a large I/O burst when cache reaches 100 percent of its capacity, which significantly affects the I/O response time. In forced flushing, system flushes the cache on priority by allocating more resources.

Cache Data Protection

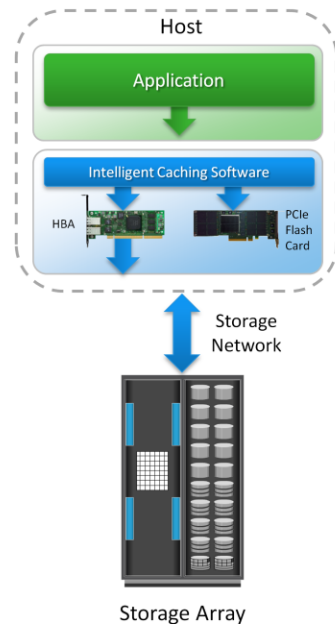
- Protects data in the cache against power or cache failures:
 - ▶ Cache mirroring
 - ▶▶ Provides protection to data against cache failure
 - ▶▶ Each write to the cache is held in two different memory locations on two independent memory cards
 - ▶ Cache vaulting
 - ▶▶ Provides protection to data against power failure
 - ▶▶ In the event of power failure, uncommitted data is dumped to a dedicated set of drives called vault drives

Cache is volatile memory, so a power failure or any kind of cache failure will cause loss of the data that is not yet committed to the disk. This risk of losing uncommitted data held in cache can be mitigated using *cache mirroring* and *cache vaulting*:

- Cache mirroring: Each write to cache is held in two different memory locations on two independent memory cards. If a cache failure occurs, the write data will still be safe in the mirrored location and can be committed to the disk. Reads are staged from the disk to the cache; therefore, if a cache failure occurs, the data can still be accessed from the disk. Because only writes are mirrored, this method results in better utilization of the available cache. In cache mirroring approaches, the problem of maintaining *cache coherency* is introduced. Cache coherency means that data in two different cache locations must be identical at all times. It is the responsibility of the array operating environment to ensure coherency.
- Cache vaulting: The risk of data loss due to power failure can be addressed in various ways: powering the memory with a battery until the AC power is restored or using battery power to write the cache content to the disk. If an extended power failure occurs, using batteries is not a viable option. This is because in intelligent storage systems, large amounts of data might need to be committed to numerous disks, and batteries might not provide power for sufficient time to write each piece of data to its intended disk. Therefore, storage vendors use a set of physical disks to dump the contents of cache during power failure. This is called *cache vaulting* and the disks are called *vault drives*. When power is restored, data from these disks is written back to write cache and then written to the intended disks.

Server Flash-caching Technology

- Uses intelligent caching software and PCIe flash card on host
 - ▶ Provides performance acceleration for read-intensive workloads
 - ▶ Avoids network latencies associated with I/O access to the storage array
- Dramatically improves application performance
 - ▶ Provides performance acceleration for read-intensive workloads
 - ▶ Avoids network latencies associated with I/O access to the storage array
- Intelligently determines data that would benefit by sitting in server on PCIe flash
- Uses minimal CPU and memory resources
 - ▶ Flash management is offloaded onto PCIe card



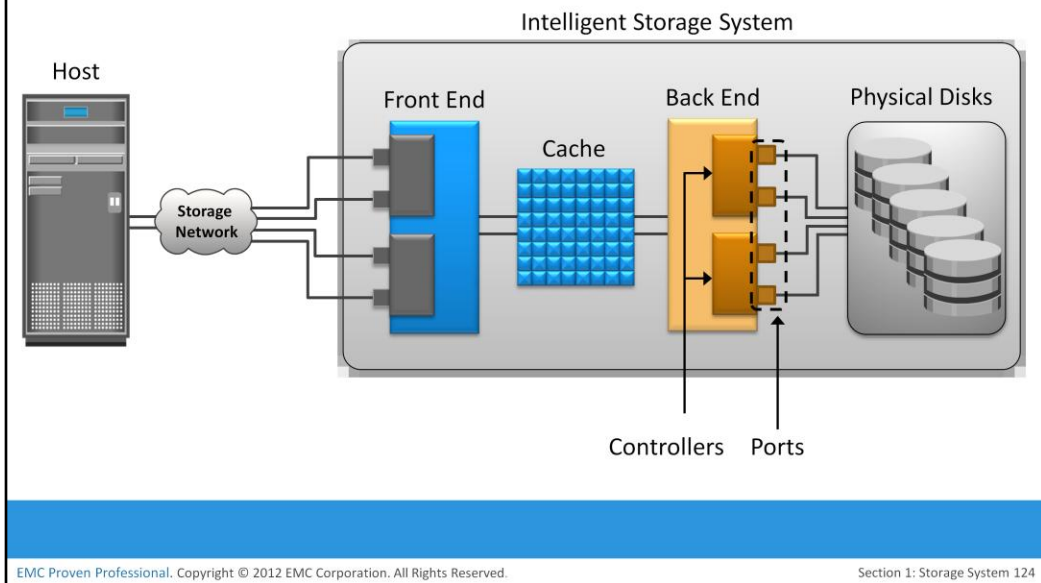
EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 123

Server flash-caching technology uses intelligent caching software and PCI Express (PCIe) flash card on the host. This dramatically improves application performance by reducing latency and accelerates throughput. Server flash-caching technology works in both physical and virtual environments and provides performance acceleration for read-intensive workloads. This technology uses minimal CPU and memory resources from the server by offloading flash management onto the PCIe card.

It intelligently determines which data would benefit by sitting in the server on PCIe flash and closer to the application. This avoids the latencies associated with I/O access over the network to the storage array. With this, the processing power required for an application's most frequently referenced data is offloaded from the back-end storage to the PCIe card. Therefore, the storage array can allocate greater processing power to other applications.

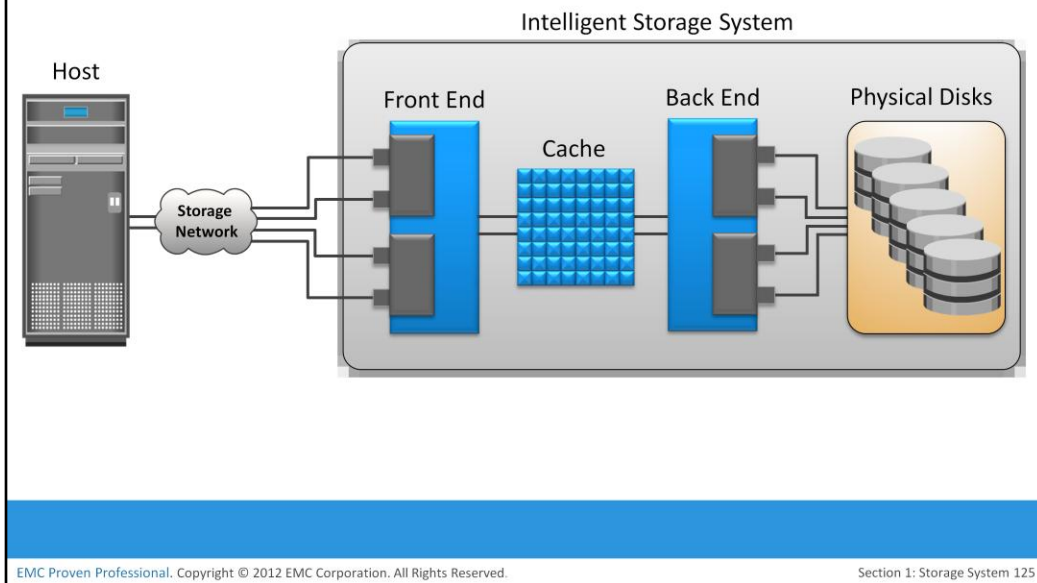
Key Components of ISS: Back End



The *back end* provides an interface between cache and the physical disks. It consists of two components: back-end ports and back-end controllers. The back-end controls data transfers between cache and the physical disks. From cache, data is sent to the back end and then routed to the destination disk. Physical disks are connected to ports on the back end. The back-end controller communicates with the disks when performing reads and writes and also provides additional, but limited, temporary data storage. The algorithms implemented on back-end controllers provide error detection and correction, along with RAID functionality.

For high data protection and high availability, storage systems are configured with dual controllers with multiple ports. Such configurations provide an alternative path to physical disks if a controller or port failure occurs. This reliability is further enhanced if the disks are also dual-ported. In that case, each disk port can connect to a separate controller. Multiple controllers also facilitate load balancing.

Key Components of ISS: Physical Disks



Physical disks are connected to the back-end storage controller and provide persistent data storage. Modern intelligent storage systems provide support to a variety of disk drives with different speeds and types, such as FC, SATA, SAS, and flash drives. They also support the use of a mix of flash, FC, or SATA within the same array.

Module 4: Intelligent Storage System

Lesson 2: Storage Provisioning and ISS Implementation

During this lesson the following topics are covered:

- Traditional storage provisioning
- Virtual storage provisioning
- ISS implementation

This lesson focuses on traditional and virtual storage provisioning. This lesson also focuses on ISS implementation.

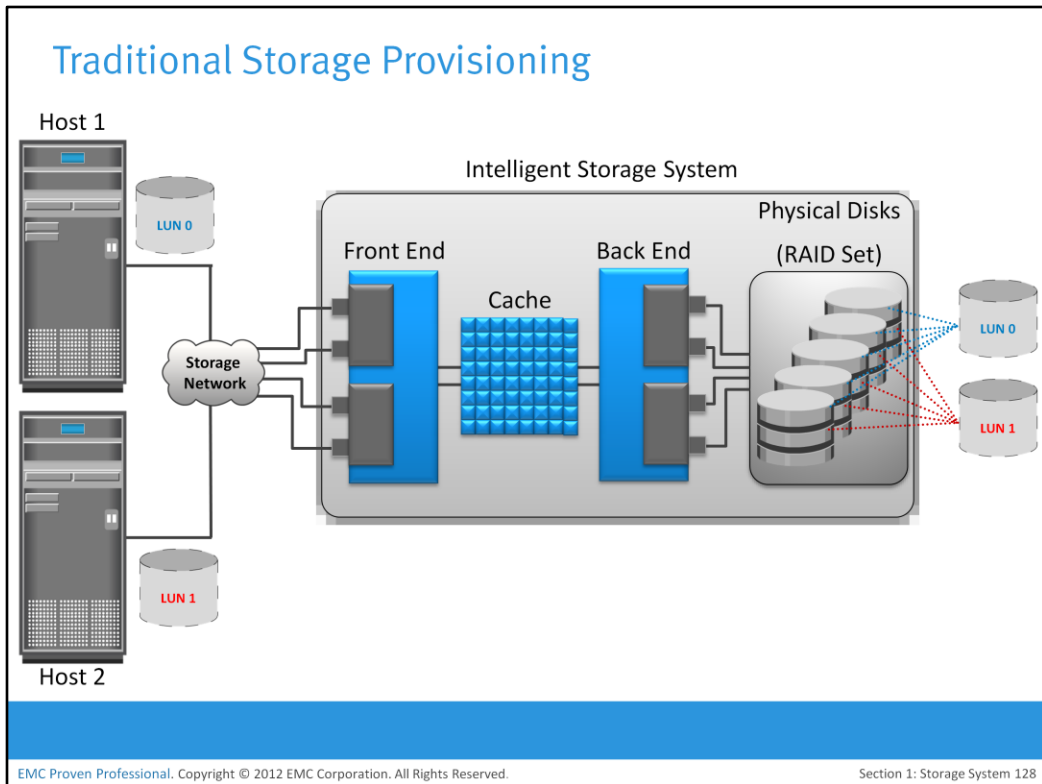
Assigning Storage to Host

Storage Provisioning

It is the process of assigning storage resources to hosts based on capacity, availability, and performance requirements of applications running on the hosts.

- Can be performed in two ways:
 - ▶ Traditional storage provisioning
 - ▶ Virtual storage provisioning

Storage provisioning is the process of assigning storage resources to hosts based on capacity, availability, and performance requirements of applications running on the hosts. Storage provisioning can be performed in two ways: traditional and virtual. *Virtual provisioning* leverages virtualization technology for provisioning storage for applications.



In traditional storage provisioning, physical disks are logically grouped together on which a required RAID level is applied to form a set, called RAID set. The number of drives in the RAID set and the RAID level determine the availability, capacity, and performance of the RAID set. It is highly recommend that the RAID set be created from drives of the same type, speed, and capacity to ensure maximum usable capacity, reliability, and consistency in performance. For example, if drives of different capacities are mixed in a RAID set, the capacity of the smallest drive is used from each disk in the set to make up the RAID set's overall capacity. The remaining capacity of the larger drives remains unused. Likewise, mixing higher revolutions per minute (RPM) drives with lower RPM drives lowers the overall performance of the RAID set.

RAID sets usually have a large capacity because they combine the total capacity of individual drives in the set. *Logical units* are created from the RAID sets by partitioning (seen as slices of the RAID set) the available capacity into smaller units. These units are then assigned to the host based on their storage requirements. Logical units are spread across all the physical disks that belong to that set. Each logical unit created from the RAID set is assigned a unique ID, called a *logical unit number* (LUN). LUNs hide the organization and composition of the RAID set from the hosts. LUNs created by traditional storage provisioning methods are also referred to as *thick LUNs* to distinguish them from the LUNs created by virtual provisioning methods.

Figure on the slide shows a RAID set consisting of five disks that have been sliced, or partitioned, into two LUNs: LUN 0 and LUN 1. These LUNs are then assigned to Host1 and Host 2 for their storage requirements.

Cont..

When a LUN is configured and assigned to a non-virtualized host, a bus scan is required to identify the LUN. This LUN appears as a raw disk to the operating system. To make this disk usable, it is formatted with a file system and then the file system is mounted.

In a virtualized host environment, the LUN is assigned to the hypervisor, which recognizes it as a raw disk. This disk is configured with the hypervisor file system, and then virtual disks are created on it. *Virtual disks* are files on the hypervisor file system. The virtual disks are then assigned to virtual machines and appear as raw disks to them. To make the virtual disk usable to the virtual machine, similar steps are followed as in a non-virtualized environment. Here, the LUN space may be shared and accessed simultaneously by multiple virtual machines.

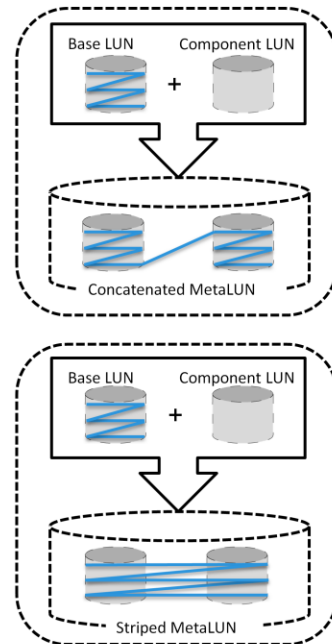
Virtual machines can also access a LUN directly on the storage system. In this method the entire LUN is allocated to a single virtual machine. Storing data in this way is recommended when the applications running on the virtual machine are response-time sensitive, and sharing storage with other virtual machines may impact their response time. The direct access method is also used when a virtual machine is clustered with a physical machine. In this case, the virtual machine is required to access the LUN that is being accessed by the physical machine.

LUN Expansion

MetaLUN

It is a method to expand LUNs that require additional capacity or performance.

- Created by combining two or more LUNs
- MetaLUNs can either be concatenated or striped
- Concatenated metaLUN
 - ▶ Provides only additional capacity but no performance
 - ▶ Expansion is quick as data is not restriped
- Striped metaLUN
 - ▶ Provides capacity and performance
 - ▶ Expansion is slow as data is restriped

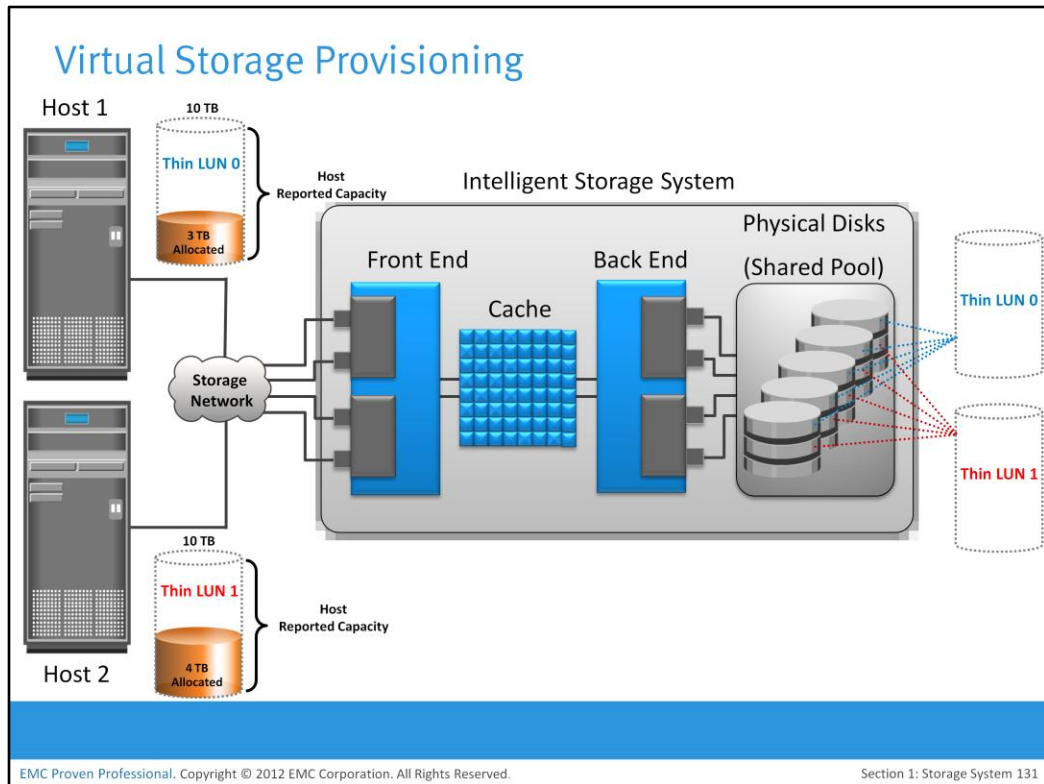


MetaLUN is a method to expand LUNs that require additional capacity or performance. A metaLUN can be created by combining two or more LUNs. A metaLUN consists of a base LUN and one or more component LUNs. MetaLUNs can be either *concatenated* or *striped*.

Concatenated expansion simply adds additional capacity to the base LUN. In this expansion, the component LUNs are not required to be of the same capacity as the base LUN. All LUNs in a concatenated metaLUN must be either protected (parity or mirrored) or unprotected (RAID 0). RAID types within a metaLUN can be mixed. For example, a RAID 1/0 LUN can be concatenated with a RAID 5 LUN. However, a RAID 0 LUN can be concatenated only with another RAID 0 LUN. Concatenated expansion is quick but does not provide any performance benefit.

Striped expansion restripes the base LUN's data across the base LUN and component LUNs. In striped expansion, all LUNs must be of the same capacity and RAID level. Striped expansion provides improved performance due to the increased number of drives being striped.

All LUNs in both concatenated and striped expansion must reside on the same disk-drive type: either all Fibre Channel or all ATA.

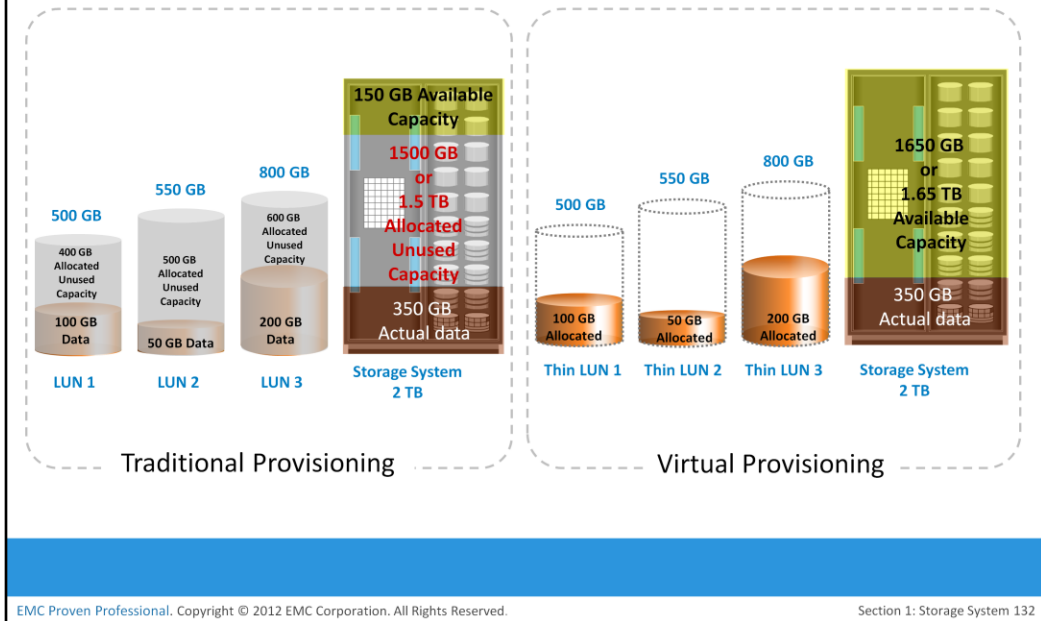


Virtual provisioning enables creating and presenting a LUN with more capacity than is physically allocated to it on the storage array. The LUN created using virtual provisioning is called a *thin LUN* to distinguish it from the traditional LUN.

Thin LUNs do not require physical storage to be completely allocated to them at the time they are created and presented to a host. Physical storage is allocated to the host “on-demand” from a shared pool of physical capacity. A *shared pool* consists of physical disks. A shared pool in virtual provisioning is analogous to a RAID group, which is a collection of drives on which LUNs are created. Similar to a RAID group, a shared pool supports a single RAID protection level. However, unlike a RAID group, a shared pool might contain large numbers of drives. Shared pools can be homogeneous (containing a single drive type) or heterogeneous (containing mixed drive types, such as flash, FC, SAS, and SATA drives).

Virtual provisioning enables more efficient allocation of storage to hosts. Virtual provisioning also enables oversubscription, where more capacity is presented to the hosts than is actually available on the storage array. Both shared pool and thin LUN can be expanded nondisruptively as the storage requirements of the hosts grow. Multiple shared pools can be created within a storage array, and a shared pool may be shared by multiple thin LUNs.

Traditional Provisioning vs. Virtual Provisioning



Administrators typically allocate storage capacity based on anticipated storage requirements. This generally results in the over provisioning of storage capacity, which then leads to higher costs and lower capacity utilization. Administrators often over-provision storage to an application for various reasons such as, to avoid frequent provisioning of storage if the LUN capacity is exhausted, and to reduce disruption to application availability. Over provisioning of storage often leads to additional storage acquisition and operational costs.

Virtual provisioning addresses these challenges. Virtual provisioning improves storage capacity utilization and simplifies storage management. Figure on the slide illustrates an example, comparing virtual provisioning with traditional storage provisioning.

With traditional provisioning, three LUNs are created and presented to one or more hosts. The total storage capacity of the storage system is 2 TB. The allocated capacity of LUN 1 is 500 GB, of which only 100 GB is consumed, and the remaining 400 GB is unused. The size of LUN 2 is 550 GB, of which 50 GB is consumed, and 500 GB is unused. The size of LUN 3 is 800 GB, of which 200 GB is consumed, and 600 GB is unused. In total, the storage system has 350 GB of data, 1.5 TB of allocated but unused capacity, and only 150 GB of remaining capacity available for other applications.

Now consider the same 2 TB storage system with virtual provisioning. Here, three thin LUNs of the same sizes are created. However, there is no allocated unused capacity. In total, the storage system with virtual provisioning has the same 350 GB of data, but 1.65 TB of capacity is available for other applications, whereas only 150 GB is available in traditional storage provisioning.

Cont..

Virtual provisioning and thin LUN offer many benefits, although in some cases traditional LUN is better suited for an application. Thin LUNs are appropriate for applications that can tolerate performance variations. In some cases, performance improvement is perceived when using a thin LUN, due to striping across a large number of drives in the pool. However, when multiple thin LUNs contend for shared storage resources in a given pool, and when utilization reaches higher levels, the performance can degrade. Thin LUNs provide the best storage space efficiency and are suitable for applications where space consumption is difficult to forecast. Using thin LUNs benefits organizations in reducing power and acquisition costs and in simplifying their storage management.

Traditional LUNs are suited for applications that require predictable performance. Traditional LUNs provide full control for precise data placement and allow an administrator to create LUNs on different RAID groups if there is any workload contention. Organizations that are not highly concerned about storage space efficiency may still use traditional LUNs.

Both traditional and thin LUNs can coexist in the same storage array. Based on the requirement, an administrator may migrate data between thin and traditional LUNs.

LUN Masking

LUN Masking

It is a process that provides data access control by defining which LUNs a host can access.

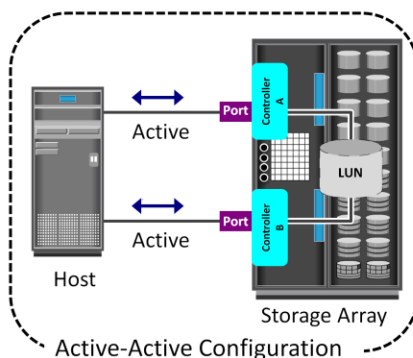
- Implemented on storage array
- Prevents unauthorized or accidental use of LUNs in a shared environment

LUN masking is a process that provides data access control by defining which LUNs a host can access. The LUN masking function is implemented on the storage array. This ensures that volume access by hosts is controlled appropriately, preventing unauthorized or accidental use in a shared environment.

For example, consider a storage array with two LUNs that store data of the sales and finance departments. Without LUN masking, both departments can easily see and modify each other's data, posing a high risk to data integrity and security. With LUN masking, LUNs are accessible only to the designated hosts.

Types of ISS: High-end Storage Systems

- Referred as active-active arrays, and generally aimed at large enterprise applications
 - ▶ Performs I/Os to LUNs through all the available paths
- These arrays provide the following capabilities:
 - ▶ Large storage capacity and cache
 - ▶ Fault tolerant architecture
 - ▶ Connectivity to mainframe and open systems
 - ▶ Multiple front-end ports and interface protocols
 - ▶ Ability to handle large amount of concurrent I/Os
 - ▶ Support local and remote data replication



EMC Proven Professional. Copyright © 2012 EMC Corporation. All Rights Reserved.

Section 1: Storage System 135

Intelligent storage systems generally fall into one of the following two categories: high-end storage systems and midrange storage systems.

High-end storage systems, referred to as *active-active arrays*, are generally aimed at large enterprise applications. These systems are designed with a large number of controllers and cache memory. An active-active array implies that the host can perform I/Os to its LUNs through any of the available controllers.

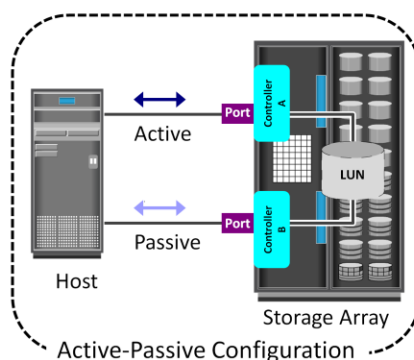
To address enterprise storage needs, these arrays provide the following capabilities:

- Large storage capacity
- Large amounts of cache to service host I/Os optimally
- Fault tolerance architecture to improve data availability
- Connectivity to mainframe computers and open systems hosts
- Availability of multiple front-end ports and interface protocols to serve a large number of hosts
- Availability of multiple back-end controllers to manage disk processing
- Scalability to support increased connectivity, performance, and storage capacity requirements
- Ability to handle large amounts of concurrent I/Os from a number of hosts and applications
- Support for array-based local and remote data replication

In addition to these features, high-end systems possess some unique features that are required for mission-critical applications.

Types of ISS: Midrange Storage Systems

- Referred as active-passive arrays, and generally aimed at small and medium-sized enterprise applications
 - ▶ Performs I/Os to LUNs only through active paths
- These arrays typically have two controllers, each with cache, RAID controllers, and disks drive interfaces
- Less front-end ports, storage capacity, and cache as compared to high-end arrays
- Support local and remote data replication



Midrange storage systems are also referred to as *active-passive arrays* and are best suited for small- and medium-sized enterprise applications. They also provide optimal storage solutions at a lower cost. In an active-passive array, a host can perform I/Os to a LUN only through the controller that owns the LUN. The host can perform reads or writes to the LUN only through the path to controller A because controller A is the owner of that LUN. The path to controller B remains passive and no I/O activity is performed through this path.

Midrange storage systems are typically designed with two controllers, each of which contains host interfaces, cache, RAID controllers, and interface to disk drives.

Midrange arrays are designed to meet the requirements of small and medium enterprise application; therefore, they host less storage capacity and cache than high-end storage arrays. There are also fewer front-end ports for connection to hosts. However, they ensure high redundancy and high performance for applications with predictable workloads. They also support array-based local and remote replication.

Module 4: Intelligent Storage System

Concept in Practice

- EMC VNX
- EMC Symmetrix VMAX

The Concept in Practice covers the product example of intelligent storage system. It covers two products: EMC Symmetrix and EMC VNX.

EMC VNX

- EMC's midrange storage offering
- Unified storage offering that provides storage for block, file, and object data
- Ideally suited for applications with predictable workloads



EMC VNX

The EMC VNX storage array is EMC's midrange storage offering that delivers enterprise-quality features and functionalities. EMC VNX is a unified storage platform that offers storage for block, file, and object-based data within the same array. It is ideally suited for applications with predictable workloads that require moderate-to-high throughput.

EMC Symmetrix VMAX

- EMC's high-end storage offering
- Key features supported by Symmetrix VMAX are:
 - ▶ Incrementally scalable to 2,400 disks
 - ▶ Supports up to 8 VMAX engines
 - ▶ Supports flash drives, fully automated storage tiering (FAST), virtual provisioning, and cloud computing
 - ▶ Supports up to 1 TB of global cache memory
 - ▶ Supports FC, iSCSI, GigE, and FICON for host connectivity
 - ▶ Supports RAID levels 1, 1+0, 5, and 6
 - ▶ Supports storage-based replication via EMC TimeFinder and SRDF



EMC Symmetrix VMAX

EMC Symmetrix establishes the highest standards for performance and capacity for an enterprise information storage solution and is recognized as the industry's most trusted storage platform. Symmetrix offers the highest level of scalability and performance to meet even unpredictable I/O workload requirements. The EMC Symmetrix offering includes Symmetrix Virtual Matrix (VMAX) series.

The EMC Symmetrix VMAX series is an innovative platform built around a scalable Virtual Matrix architecture to support the future storage growth demands of virtual IT environments. The key features supported by Symmetrix VMAX follows:

- Incrementally scalable to 2,400 disks
- Supports up to 8 VMAX engines (Each VMAX engine contains a pair of directors)
- Supports flash drives, fully automated storage tiering (FAST), virtual provisioning, and Cloud computing
- Supports up to 1 TB of global cache memory
- Supports FC, iSCSI, GigE, and FICON for host connectivity
- Supports RAID levels 1, 1+0, 5, and 6
- Supports storage-based replication through EMC TimeFinder and EMC SRDF
- Highly fault-tolerant design that allows nondisruptive upgrades and full component-level redundancy with hot-swappable replacements

Module 4: Summary

Key points covered in this module:

- Key components of intelligent storage system
- Cache management and protection techniques
- Storage provisioning methods
- Types of intelligent storage systems

This module covered the four key components of intelligent storage systems, front end, cache, back end, and physical disks. Further, this module detailed the cache management and protection techniques such as flushing, least recently used, and most recently used algorithm. It also covered the two storage provisioning techniques, traditional and virtual storage provisioning. Finally, the module described the two types of intelligent storage systems such as high-end and midrange storage systems.

Check Your Knowledge – 1

- Which component of an intelligent storage system isolates host from the mechanical delays associated with rotating disks?
 - A. Front-end controller
 - B. Back-end controller
 - C. Cache
 - D. Storage network
- Which mode of flushing is activated when the cache reaches 100% of its capacity?
 - A. Idle
 - B. High watermark
 - C. Forced
 - D. Low watermark

Check Your Knowledge – 2

- In traditional storage provisioning, which LUN expansion technique provides improved performance?
 - A. Concatenated metaLUN
 - B. Striped metaLUN
 - C. Base LUN
 - D. Component LUN
- Which process provides data access control by restricting host access to specific LUN(s)?
 - A. LUN masking
 - B. Zoning
 - C. Trespassing
 - D. VSAN

Check Your Knowledge – 3

- Which mechanism provides protection to 'uncommitted data in cache' against power failure?
 - A. Mirroring
 - B. Vaulting
 - C. Watermarking
 - D. Tiering

Section 1: Summary

Key points covered in this section:

- Data and information
- Core elements of data center
- Virtualization and cloud computing
- Disk drive components and performance
- Common RAID levels
- Key components of intelligent storage system

This section covered types of data and five core elements of a data center – application, DBMS, host, network, and storage. This section also provided an overview on virtualization and cloud computing. It also covered disk drive components and factors governing disk drive performance. Further, this section covered commonly used RAID levels such as 0, 1, 1+0, 3, 5, and 6. Finally, it covered the key components of intelligent storage systems – front end, cache, back end, and physical disks along with implementation of intelligent storage system.