Chapter 10
Storage Virtualization

KEY CONCEPTS

- Memory Virtualization
- Network Virtualization
- Server Virtualization
- Storage Virtualization
- In-Band and Out-of-Band Implementations
- Block-Level and File-Level Virtualization

As storage networking technology matures, larger and complex implementations are becoming more common. The heterogeneous nature of storage infrastructures has further added to the complexity of managing and utilizing storage resources effectively. Specialized technologies are required to meet stringent service level agreements and to provide an adaptable infrastructure with reduced cost of management. The virtualization technologies discussed in this chapter provide enhanced productivity, asset utilization, and better management of the storage infrastructure.

Virtualization is the technique of masking or abstracting physical resources, which simplifies the infrastructure and accommodates the increasing pace of business and technological changes. It increases the utilization and capability of IT resources, such as servers, networks, or storage devices, beyond their physical limits. Virtualization simplifies resource management by pooling and sharing resources for maximum utilization and makes them appear as logical resources with enhanced capabilities.

10.1 Forms of Virtualization

Virtualization has existed in the IT industry for several years and in different forms, including memory virtualization, network virtualization, server virtualization, and storage virtualization.

10.1.1 Memory Virtualization

Virtual memory makes an application appear as if it has its own contiguous logical memory independent of the existing physical memory resources.

Since the beginning of the computer industry, memory has been and continues to be an expensive component of a host. It determines both the size and the number of applications that can run on a host.

With technological advancements, memory technology has changed and the cost of memory has decreased. Virtual memory managers (VMMs) have evolved, enabling multiple applications to be hosted and processed simultaneously.

In a virtual memory implementation, a memory address space is divided into contiguous blocks of fixed-size pages. A process known as *paging* saves inactive memory pages onto the disk and brings them back to physical memory when required. This enables efficient use of available physical memory among different processes. The space used by VMMs on the disk is known as a *swap file*. A swap file (also known as *page file* or *swap space*) is a portion of the hard disk that functions like physical memory (RAM) to the operating system. The operating system typically moves the least used data into the swap file so that RAM will be available for processes that are more active. Because the space allocated to the swap file is on the hard disk (which is slower than the physical memory), access to this file is slower.

10.1.2 Network Virtualization

Network virtualization creates virtual networks whereby each application sees its own logical network independent of the physical network. A *virtual LAN (VLAN)* is an example of network virtualization that provides an easy, flexible, and less expensive way to manage networks. VLANs make large networks more manageable by enabling a centralized configuration of devices located in physically diverse locations.

Consider a company in which the users of a department are separated over a metropolitan area with their resources centrally located at one office. In a typical network, each location has its own network connected to the others through routers. When network packets cross routers, latency influences network performance. With VLANs, users with similar access requirements can be grouped together into the same virtual network. This setup eliminates the need for network routing. As a result, although users are physically located at disparate locations, they appear to be at the same location accessing resources locally. In addition to improving network performance, VLANs also provide enhanced security by isolating sensitive data from the other networks and by restricting access to the resources located within the networks.

Virtual SAN (VSAN)

A *virtual SAN/virtual fabric* is a recent evolution of SAN and conceptually, functions in the same way as a VLAN.

In a VSAN, a group of hosts or storage ports communicate with each other using a virtual topology defined on the physical SAN. VSAN technology enables users to build one or more Virtual SANs on a single physical topology containing switches and ISLs. This technology improves storage area network (SAN) scalability, availability, and security. These benefits are derived from the separation of Fibre Channel services in each VSAN and isolation of traffic between VSANs. Some of the features of VSAN are:

- Fibre Channel ID (FC ID) of a host in a VSAN can be assigned to a host in another VSAN, thus improving scalability of SAN.
- Every instance of a VSAN runs all required protocols such as FSPF, domain manager, and zoning.
- Fabric-related configurations in one VSAN do not affect the traffic in another VSAN.
- Events causing traffic disruptions in one VSAN are contained within that VSAN and are not propagated to other VSANs.

10.1.3 Server Virtualization

Server virtualization enables multiple operating systems and applications to run simultaneously on different v*irtual machines* created on the same physical server (or group of servers). Virtual machines provide a layer of abstraction between the operating system and the underlying hardware. Within a physical server, any number of virtual servers can be established; depending on hardware capabilities (see Figure 10-1). Each virtual server seems like a physical machine to the operating system, although all virtual servers share the same underlying physical hardware in an isolated manner. For example, the physical memory is shared between virtual servers but the address space is not. Individual virtual servers can be restarted, upgraded, or even crashed, without affecting the other virtual servers on the same physical machine.

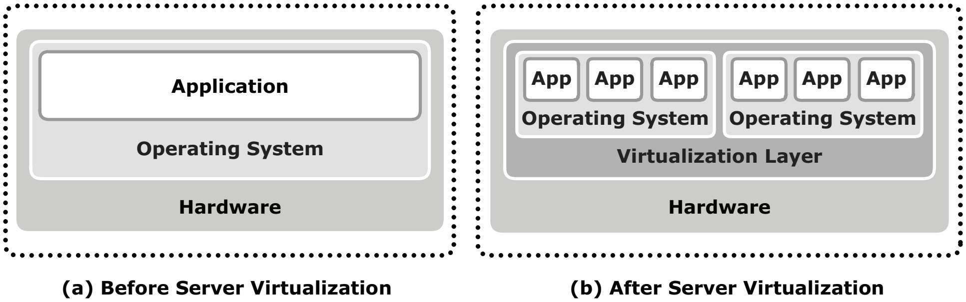

Figure 10-1: Server virtualization

With changes in computing from a dedicated to a client/server model, the physical server faces resource conflict issues when two or more applications running on these servers have conflicting requirements (e.g., need different values in the same registry entry, different versions of the same DLL). 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, as shown in Figure 10-1(a). 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.

Server virtualization addresses the issues that exist in a physical server environment. The virtualization layer, shown in Figure 10-1(b), helps to overcome resource conflicts by isolating applications running on different operating systems on the same machine. In addition, server virtualization can dynamically move the underutilized hardware resources to a location where they are needed most, improving utilization of the underlying hardware resources.

10.1.4 Storage Virtualization

Storage virtualization is the process of presenting a logical view of the physical storage resources to a host. This logical storage appears and behaves as physical storage directly connected to the host. Throughout the evolution of storage technology, some form of storage virtualization has been implemented. Some examples of storage virtualization are host-based volume management, LUN creation, tape storage virtualization, and disk addressing (CHS to LBA).

The key benefits of storage virtualization include increased storage utilization, adding or deleting storage without affecting an application's availability, and nondisruptive data migration (access to files and storage while migrations are in progress).

Figure 10-2 illustrates a virtualized storage environment. At the top are four servers, each of which has one virtual volume assigned, which is currently in use by an application. These virtual volumes are mapped to the actual storage in the arrays, as shown at the bottom of the figure. When I/O is sent to a virtual volume, it is redirected through the virtualization at the storage network layer to the mapped physical array.

The discussion that follows provides details about the different types of storage virtualization, methods of implementation, the challenges associated with the implementation of storage virtualization, and examples of implementation.

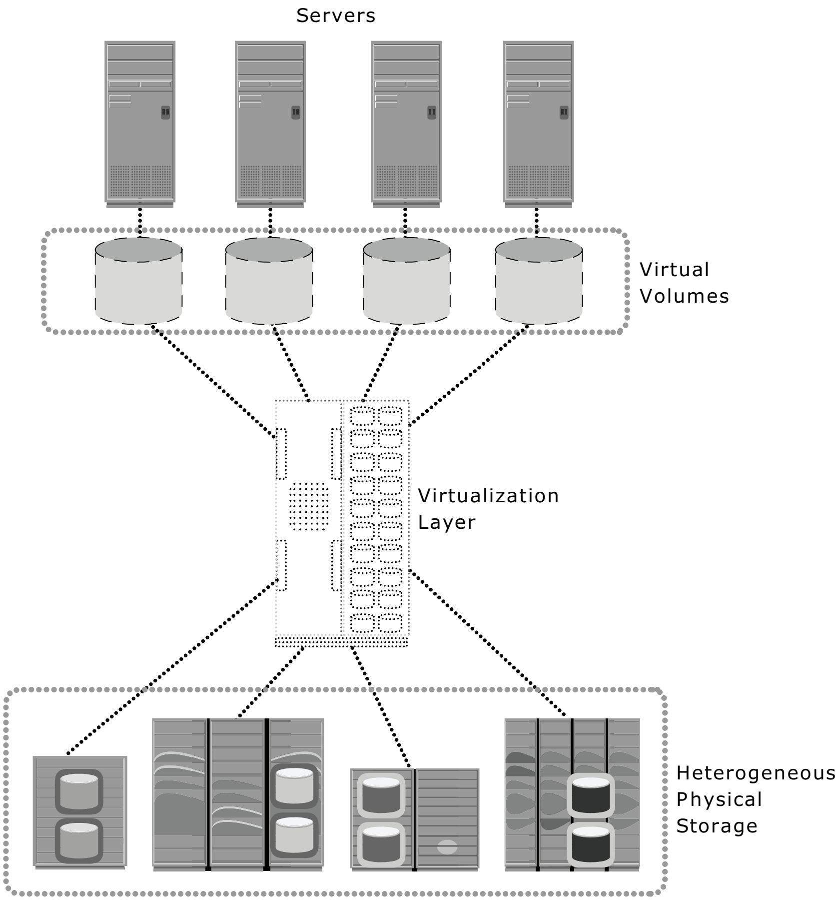

Figure 10-2: Storage virtualization

10.2 SNIA Storage Virtualization Taxonomy

The SNIA (Storage Networking Industry Association) storage virtualization taxonomy (see Figure 10-3) provides a systematic classification of storage virtualization, with three levels defining what, where, and how storage can be virtualized.

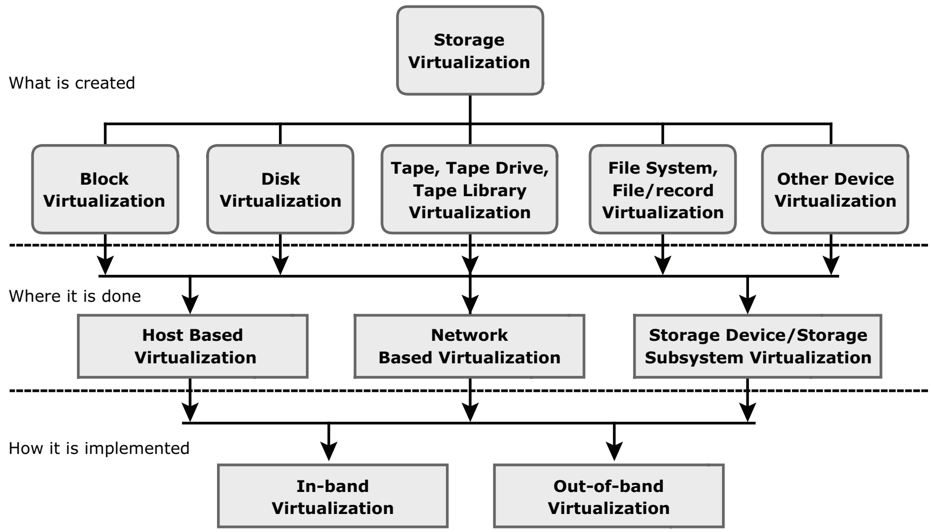

Figure 10-3: SNIA storage virtualization taxonomy

The first level of the storage virtualization taxonomy addresses "what" is created. It specifies the types of virtualization: block virtualization, file virtualization, disk virtualization, tape virtualization, or any other device virtualization. Block-level and file-level virtualization are the core focus areas covered later in this chapter.

The second level describes "where" the virtualization can take place. This requires a multilevel approach that characterizes virtualization at all three levels of the storage environment: server, storage network, and storage, as shown in Figure 10-4. An effective virtualization strategy distributes the intelligence across all three levels while centralizing the management and control functions. Data storage functions—such as RAID, caching, checksums, and hardware scanning—should remain on the array. Similarly, the host should control application-focused areas, such as clustering and application failover, and volume management of raw disks. However, path redirection, path failover, data access, and distribution or load-balancing capabilities should be moved to the switch or the network.

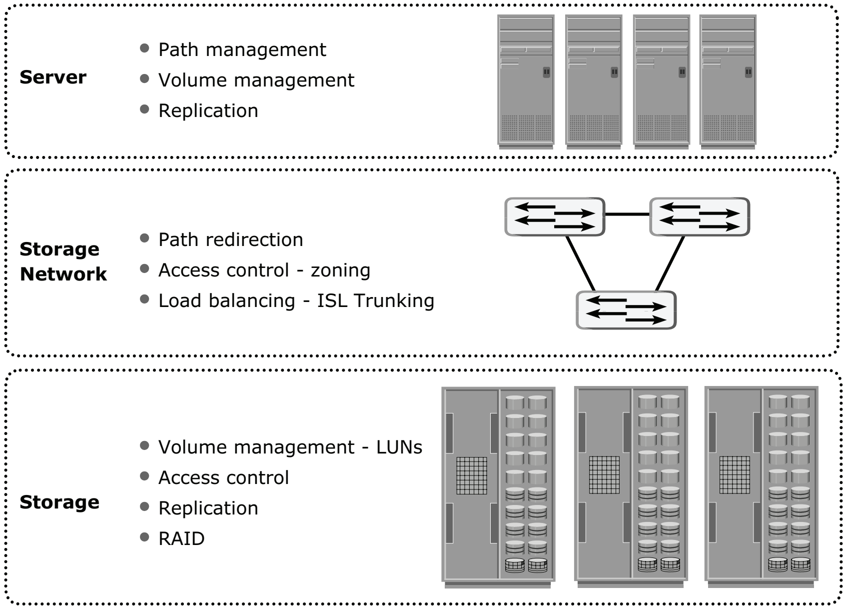

Figure 10-4: Storage virtualization at different levels of the storage environment

The third level of the storage virtualization taxonomy specifies the network level virtualization methodology, in-band or out-of-band.

10.3 Storage Virtualization Configurations

Storage virtualization at the network is implemented using either the in-band or the out-of-band methodology. In an *out-of-band* implementation, the virtualized environment configuration is stored external to the data path. As shown in Figure 10-5(a), the configuration is stored on the virtualization appliance configured external to the storage network that carries the data. This configuration is also called split-path because the control and data paths are split (the control path runs through the appliance, the data path does not). This configuration enables the environment to process data at a network speed with only minimal latency added for translation of the virtual configuration to the physical storage. The data is not cached at the virtualization appliance beyond what would normally occur in a typical SAN configuration. Since the virtualization appliance is hardware-based and optimized for Fibre Channel communication, it can be scaled significantly. In addition, because the data is unaltered in an out-of-band implementation, many of the existing array features and functions can be utilized in addition to the benefits provided by virtualization.

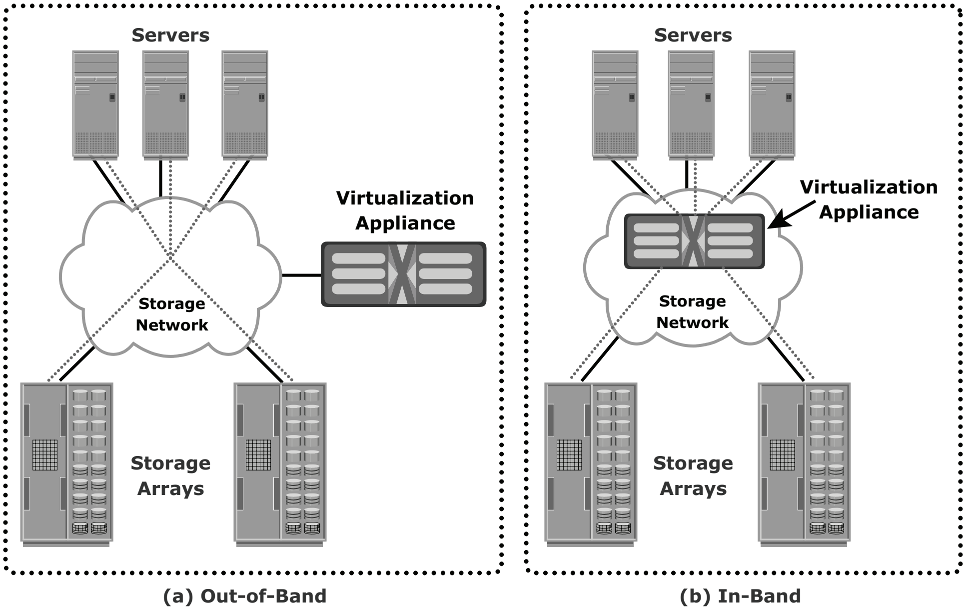

Figure 10-5: Storage virtualization configuration

The *in-band* implementation places the virtualization function in the data path, as shown in Figure 10-5(b). General-purpose servers or appliances handle the virtualization and function as a translation engine for the virtual configuration to the physical storage. While processing, data packets are often cached by the appliance and then forwarded to the appropriate target. An in-band implementation is software-based and data storing and forwarding through the appliance results in additional latency. It introduces a delay in the application response time because the data remains in the network for some time before being committed to disk.

In terms of infrastructure, the in-band architecture increases complexity and adds a new layer of virtualization (the appliance), while limiting the ability to scale the storage infrastructure. An in-band implementation is suitable for static environments with predictable workloads.

10.4 Storage Virtualization Challenges

Storage networking and feature-rich intelligent storage arrays have addressed and provided specific solutions to business problems. As an enabler, virtualization should add value to the existing solution, but introducing virtualization into an environment adds new challenges. The storage virtualization solution must be capable of addressing issues such as scalability, functionality, manageability, and support.

10.4.1 Scalability

Consider the scalability of an environment with no virtualization. This environment may have several storage arrays that provide storage independently of each other. Each array is managed independently and meets application requirements in terms of IOPS and capacity. After virtualization, a storage array can no longer be viewed as an individual entity. The environment as a whole must now be analyzed. As a result, the infrastructure that is implemented both at a physical level and from a virtualization perspective must be able to adequately handle the workload, which may consist of different types of processing and traffic distribution. Greater care must be exercised to ensure that storage devices are performing to meet the appropriate requirements.

10.4.2 Functionality

Functionality is another challenge in storage virtualization. Currently, the storage array provides a wide range of advanced functionality necessary for meeting an application's service levels. This includes local replication, extended-distance remote replication and the capability to provide application consistency across multiple volumes and arrays. In a virtualized environment, the virtual device must provide the same or better functionality than what is currently available on the storage array, and it must continue to leverage existing functionality on the arrays. It should protect the existing investments in processes, skills, training, and human resources.

10.4.3 Manageability

The management of the storage infrastructure in a virtualized environment is an important consideration for storage administrators. A key advantage of today's storage resource management tools in an environment without virtualization is that they provide an end-to-end view, which integrates all the resources in the storage environment. They provide efficient and effective monitoring, reporting, planning, and provisioning services to the storage environment.

Introducing a virtualization device breaks the end-to-end view into three distinct domains: the server to the virtualization device, the virtualization device to the physical storage, and the virtualization device itself. The virtualized storage environment must be capable of meeting these challenges and must integrate with existing management tools to enable management of an end-to-end virtualized environment.

10.4.4 Support

Virtualization is not a stand-alone technology but something that has to work within an existing environment. This environment may include multiple vendor technologies, such as switch and storage arrays, adding to complexity. Addressing such complexities often requires multiple management tools and introduces interoperability issues. Without a virtualization solution, many companies try to consolidate products from a single vendor to ease these challenges. Introducing a virtualization solution reduces the need to standardize on a single vendor. However, supportability issues in a virtualized heterogeneous environment introduce challenges in coordination and compatibility of products and solutions from different manufacturers and vendors.

10.5 Types of Storage Virtualization

Virtual storage is about providing logical storage to hosts and applications independent of physical resources. Virtualization can be implemented in both SAN and NAS storage environments. In a SAN, virtualization is applied at the block level, whereas in NAS, it is applied at the file level.

10.5.1 Block-Level Storage Virtualization

Block-level storage virtualization provides a translation layer in the SAN, between the hosts and the storage arrays, as shown in Figure 10-6. Instead of being directed to the LUNs on the individual storage arrays, the hosts are directed to the virtualized LUNs on the virtualization device. The virtualization device translates between the virtual LUNs and the physical LUNs on the individual arrays. This facilitates the use of arrays from different vendors simultaneously, without any interoperability issues. For a host, all the arrays appear like a single target device and LUNs can be distributed or even split across multiple arrays.

Block-level storage virtualization extends storage volumes online, resolves application growth requirements, consolidates heterogeneous storage arrays, and enables transparent volume access. It also provides the advantage of nondisruptive data migration.

In traditional SAN environments, LUN migration from one array to another was an offline event because the hosts needed to be updated to reflect the new array configuration. In other instances, host CPU cycles were required to migrate data from one array to the other, especially in a multi vendor environment. With a block-level virtualization solution in place, the virtualization engine handles the back-end migration of data, which enables LUNs to remain online and accessible while data is being migrated. No physical changes are required because the host still points to the same virtual targets on the virtualization device. However, the mappings on the virtualization device should be changed. These changes can be executed dynamically and are transparent to the end user.

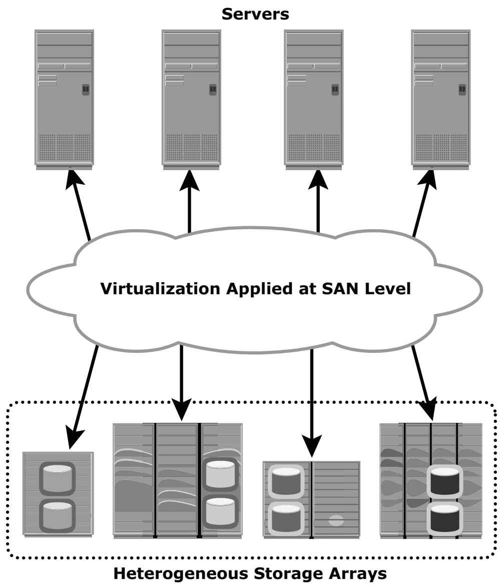

Figure 10-6: Block-level storage virtualization

Deploying heterogeneous arrays in a virtualized environment facilitates an information lifecycle management (ILM) strategy, enabling significant cost and resource optimization. Low-value data can be migrated from high- to low-performance arrays or disks. Detailed implementation of functionality and operation of block-level storage virtualization is discussed in the section "Concepts in Practice" later in this chapter.

10.5.2 File-Level Virtualization

File-level virtualization addresses the NAS challenges by eliminating the dependencies between the data accessed at the file level and the location where the files are physically stored. This provides opportunities to optimize storage utilization and server consolidation and to perform nondisruptive file migrations.

Figure 10-7 illustrates a NAS environment before and after the implementation of file-level virtualization.

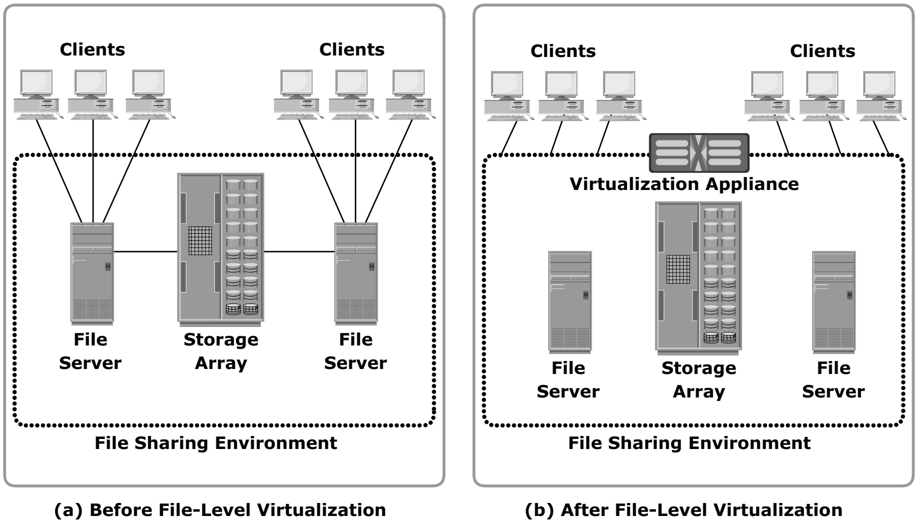

Figure 10-7: NAS device before and after file-level virtualization

Before virtualization, each NAS device or file server is physically and logically independent. Each host knows exactly where its file-level resources are located. Underutilized storage resources and capacity problems result because files are bound to a specific file server. It is necessary to move the files from one server to another because of performance reasons or when the file server fills up. Moving files across the environment is not easy and requires downtime for the file servers. Moreover, hosts and applications need to be reconfigured with the new path, making it difficult for storage administrators to improve storage efficiency while maintaining the required service level.

File-level virtualization simplifies file mobility. It provides user or application independence from the location where the files are stored. File-level virtualization creates a logical pool of storage, enabling users to use a logical path, rather than a physical path, to access files. File-level virtualization facilitates the movement of file systems across the online file servers. This means that while the files are being moved, clients can access their files non-disruptively. Clients can also read their files from the old location and write them back to the new location without realizing that the physical location has changed. Multiple clients connected to multiple servers can perform online movement of their files to optimize utilization of their resources. A global namespace can be used to map the logical path of a file to the physical path names. Detailed implementation of functionality and operation of file-level storage virtualization is discussed in the next section.

10.6 Concepts in Practice

EMC Invista and Rainfinity are EMC product implementations of block-level and file-level virtualization, respectively. These virtualization solutions offer improvements over traditional device-level controls in the area of capacity utilization, storage tier management, performance optimization, and data protection. For more details on Invista and Rainfinity, please refer to `http://education.EMC.com/ismbook`.

10.6.1 EMC Invista

EMC Invista is an out of band SAN-based block-level storage virtualization solution. It uses intelligent SAN switches with customized hardware to virtualize physical storage in a logical presentation. These switches are capable of handling data operations at network speed. They use specialized software to examine the port, logical volume, and offset to which the I/O is sent and can control the target path of I/Os to the storage devices.

Invista is physically located between the production hosts and the storage arrays, as shown in Figure 10-8. The part of Invista that is connected to the hosts is called the *front end.* The part that is connected to the storage arrays is called the *back end*. The hosts and storage are connected to the Invista hardware directly or through a SAN switch. The host and storage array connections are Fibre Channel interfaces on intelligent Fibre Channel switches within Invista.

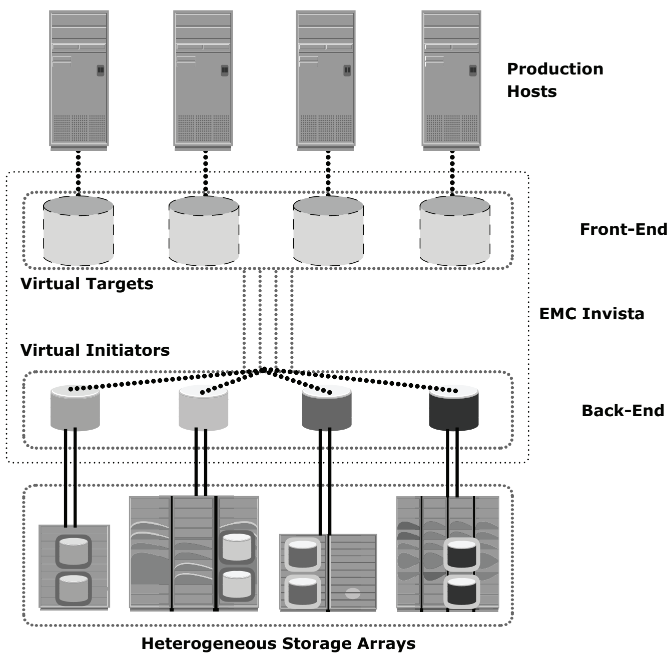

Figure 10-8: Storage virtualization with EMC Invista

Hosts see Invista as a storage device or a virtual target, whereas storage sees Invista as a host or a virtual initiator. The virtual targets are abstract entities, which are created by designating specific ports on the switch to be used as front-end ports, which become visible in the name server on the switch. Invista uses virtual targets and virtual initiators to map virtual volumes to the physical storage on back-end arrays. Invista serves as a proxy device, intercepting communications between the host and the storage by providing virtualization.

Invista Components

Figure 10-9 shows the hardware components of an Invista instance. The main hardware components are the control path cluster (CPC), the data path controller (DPC), and the Ethernet switch.

A *CPC* is a customized storage device (A dual node cluster in an active-active configuration) running Invista software. The CPC does not contain any user data; instead, it stores Invista configuration parameters, including storage device information, virtual volume information, the clone group, and information about the storage volumes belonging to the storage devices. It also performs all the control and management functions of the virtual storage.

The DPC is a special purpose SAN switch/blade. It runs special firmware and layered software that enables the creation and management of virtual initiators and targets. The DPC receives I/O from the host initiator and controls its attributes, such as target, LUN, and offset within the logical unit. The DPC performs I/O-mapping operations and redirection for read and write operations between the hosts (front end) and the storage arrays (back end). The DPC gets its configuration from the CPC.

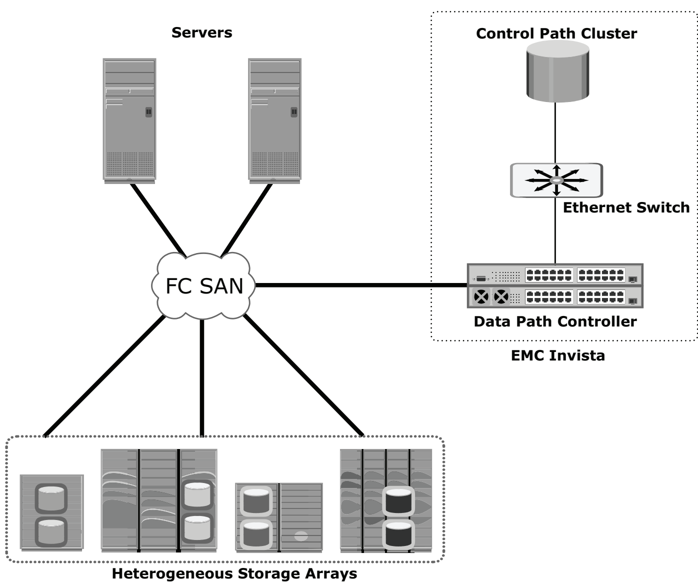

Figure 10-9: Invista's hardware components

An *Ethernet switch* connects the CPC and the DPC through a private IP network for configuration and control path traffic. The software provides dynamic volume mobility, network-based volume management, and heterogeneous point-in-time copies.

Invista Operation

When an I/O request from a host arrives at the DPC, it handles the I/O and maps it to the appropriate virtual target (or initiator), as shown in Figure 10-10. In some exceptional cases, if the command is a SCSI inquiry about the device or an I/O for which the DPC does not have mapping information, the CPC handles the request.

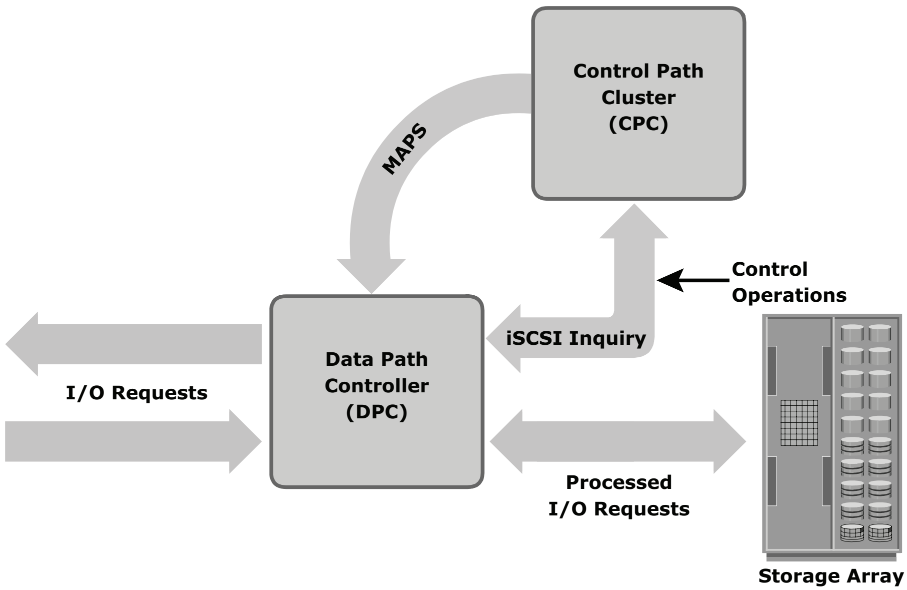

Figure 10-10: Invista operations

If a new storage array is added to the intelligent switch, the CPC discovers the new array and updates the mapping information to put that new array into use. With the mapping done, the I/O gets redirected to the new storage location.

Similarly, if an old array needs to be removed, the CPC issues another set of instructions to move the data from that old array to another array. The DPC copies the data online and the old array can be moved out nondisruptively.

Invista Advantages

EMC Invista provides block-level storage virtualization in heterogeneous storage environments. It also supports dynamic volume mobility for volume extension and data migration between different storage tiers without any downtime.

Invista supports local and remote replication functionality; and it integrates with the existing SAN infrastructure and uses the full fabric bandwidth for high-speed I/O processing. Invista provides separate data and control paths for easy management and faster I/O processing.

10.6.2 Rainfinity

Rainfinity is a dedicated hardware/software solution for file-level virtualization. The Rainfinity *Global File Virtualization (GFV)* appliance (see Figure 10-11) provides an abstraction of file-based storage transparently to users. Files can be moved from one file server to another even when clients are reading and writing their data.

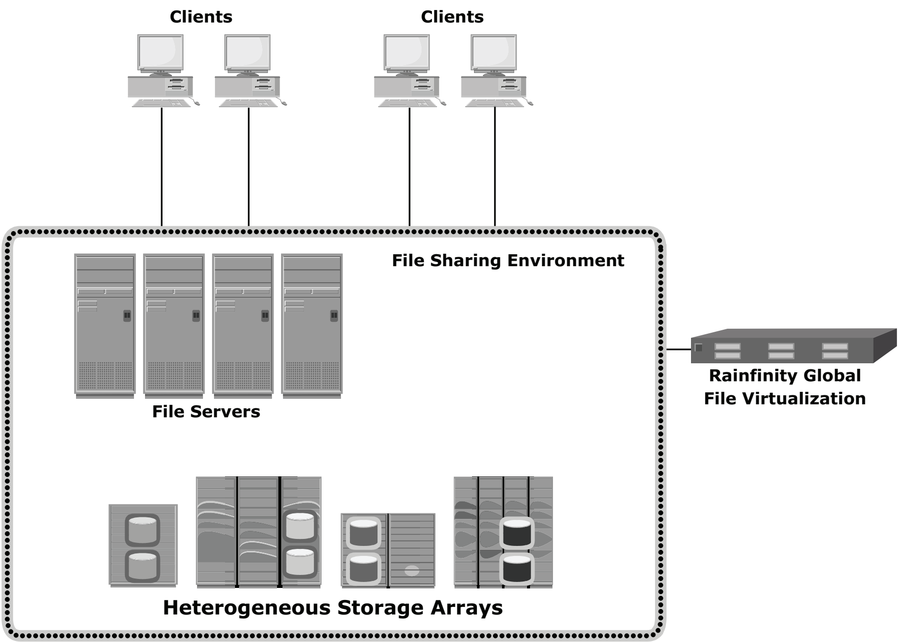

Figure 10-11: File-level virtualization with Rainfinity

A Rainfinity global namespace transparently maps the logical path names to the physical locations after the files have been moved. Therefore, users and applications are redirected to the new location without reconfiguring the physical path names. The management of the namespace can be accomplished by industry standard protocols and mechanisms, such as a Distributed File System (DFS), NIS, and LDAP. Rainfinity integrates itself with these existing industry standard namespaces.

The Rainfinity appliance integrates into the existing IP network and acts like a layer 2 bridge between the client and the file server. This enables Rainfinity to see and process the traffic between clients and file servers with minimal modification to the existing network. Rainfinity is aware of file-sharing protocols (CIFS and NFS). This application-layer intelligence enables Rainfinity to move data from one server to another without interrupting client access.

Rainfinity Components

The Rainfinity GFV appliance is a 64-bit processor with up to 16 GB of cache memory. The GFV appliance consists of two hot-swappable SCSI hard drives configured with RAID 1 to buffer all writes to the disk. It also contains a keyboard, a mouse, and a CD-ROM drive for software uploads.

Rainfinity is shipped with Rainfinity code, Windows Proxy service, and the Security ID (SID) translator. The Rainfinity code is a customized Linux-based operating system. The Windows Proxy service is installed on a separate Windows server and is required to move CIFS data. Rainfinity connects to a computer running Windows Proxy and uses it to collect performance statistics and execute administration tasks.

Rainfinity translates the security properties of the files and directories involved in a CIFS transaction with the help of the SID translator. The SID translator runs on a separate Windows server. This capability is used to assist data migrations when the access control list (ACL) is defined in terms of local groups on the source file server. When the data is migrated to the destination server, the ACL should be defined in terms of the corresponding local groups on the destination server. The rules governing such translation are defined in the SID translation tables.

Rainfinity Operations

In a NAS environment, the file servers and Rainfinity appliance are connected over an IP network. Rainfinity requires a separate VLAN in the network so that it does not interfere with the data path and clients can continue to access the storage with no disruption.

When data needs to be relocated for cost or performance optimization, the ports associated with the file servers involved in relocation are then associated with the Rainfinity VLAN. Rainfinity is in the data path for these file servers and all I/Os associated with these file servers pass through it. As Rainfinity now has control of this traffic, it can move the file system to its new location, transparently to the clients. Once the data relocation is complete, Rainfinity can update the global namespace; and the namespace, in turn, updates the clients. This update of the client namespace informs the clients about the new file system location. As clients are updated, their I/Os are now directed to the new location, removing Rainfinity from the I/O path. The new copy of the data is at the new location, and the original source reflects a point-in-time copy at the end of the data relocation.

Rainfinity treats data relocation as a transaction and has the capability to roll back transactions. During a transaction, updates to data are synchronized across the source and the new destination, eliminating the risk of data corruption. Rainfinity has an auto-complete feature that provides policy-based control on transaction completion. These policies can be framed based on the percentage of clients remapping. Rainfinity can handle multiple simultaneous transactions but performs only one active move transaction at a time, queuing up other transactions. Once the initial data copy is accomplished, multiple switching transactions are allowed.

Rainfinity uses the best characteristics of both the in-band and out-of-band method. The Rainfinity appliance remains out of band until it is required for data mobility. When Rainfinity is not performing any move or redirecting access task, all the file servers remain in the public LAN segment. When files are moved, the two file servers involved in the move must be part of the Rainfinity LAN segment (VLAN) and Rainfinity comes in-band.

Global Namespace Management

Rainfinity *Global Namespace Appliance* (GNA) allows storage administrators to remove the physical attributes associated with file storage and introduce a logical namespace in their environment. With a scalable, transparent file protocol-switching capability, a global namespace stores namespace schemes, provides directory services, and controls the file access point of CIFS and NFS clients. Commonly used operating systems include the client and server global namespace software that dynamically manages client referral and local mounts. The GFV *Global Namespace Management Application* provides an interface to view and manage file system namespaces. This application presents a unified view of global namespaces so that it is easier to understand the logical structure of the files. In addition, the application centrally manages distributed global namespaces by subscribing to and publishing namespace schemas that are stored on the DFS, NIS, and LDAP servers. The published namespaces are used by other Rainfinity applications that relocate data, while providing continuous read/write access.

The unified namespace view enables the creation of a multiprotocol global namespace that is serviced by independent CIFS and NFS global namespace servers. The Global Namespace Management Application merges the contents of namespace schemas by matching logical names, presenting a unified namespace hierarchy. In addition, the migration and consolidation application automatically updates the physical locations in both namespace schemas. These multiprotocol namespace synchronization capabilities eliminate the manual administrative tasks of maintaining separate namespaces.

Rainfinity Advantages

Like Invista, Rainfinity offers capacity management and storage consolidation. Rainfinity also provides tiered storage management support to achieve the enterprise ILM strategy. Rainfinity's primary application and advantage is transparent data mobility.

Summary

Virtualization provides flexibility while easing management of the existing infrastructure. Virtualization enables users to optimally utilize current processes, technologies, and systems. It allows for the addition, modification, or replacement of physical resources without affecting application availability. Virtualization technology offers high security and data integrity, which are mandatory for centralized computing environments. It also reduces performance degradation issues and unplanned downtime due to faults, and ensures increased availability of hardware resources.

This chapter detailed the different forms of virtualization and their benefits. It also covered block-level and file-level storage virtualization and provided associated product examples, explaining their processes. The data mobility features in virtualization ensure uninterrupted storage operation and prevent application outages due to any resource conflict or unavailability.

Resources and data are still vulnerable to natural disasters and other planned and unplanned outages, which can affect data availability. The next chapter covers business continuity and describes disaster recovery solutions that ensure high availability and uninterrupted business operations.

EXERCISES

1. **What do VLANs virtualize? Discuss VLAN implementation as a virtualization technology.**
2. **Research SNIA's storage virtualization taxonomy and write a short technical note.**
3. **How can a block-level virtualization implementation be used as a data migration tool? Explain how data migration will be accomplished and discuss the advantages of using this method for storage. Compare this method to traditional migration methods.**
4. **Frequently, storage arrays in a data center are replaced with newer arrays to take advantage of technology advancements and cost benefits and to allow business growth. Migrating data from old arrays to a new array has now become a routinely performed activity in data centers. Do a survey of host-based, storage array–based, and virtualization appliance–based migration methods. Detail the advantages and disadvantages. Consider a migration scenario in which you are migrating from a DAS to a SAN environment.**
5. **Refer to question 4. Which method of migration will you use? Develop a short presentation explaining why you are recommending a particular method. Include a work breakdown structure for executing the migration with your recommended method.**