Chapter 2 Preparing for Server Installation

In this chapter, you will:

- ♦ Identify server categories
- ♦ Evaluate server components
- ♦ Discuss server performance
- ♦ Plan for system disasters and reduce their effects
- ♦ Evaluate network components
- ♦ IP addressing
- ♦ **Download and Install VMware Player** and discuss Virtual Machine Settings and VMware Tools

In Chapter 1, you learned about network components, with an emphasis on the Web server and related server products. In this chapter, you will focus on other server components and on virtualization. Because a server malfunction affects many people, you need to minimize server problems, typically through duplication of hardware. You will learn how server components affect performance and how servers communicate with clients in a Web environment and on the Internet. Finally, you will learn about configuring IP addressing for the servers.

IDENTIFYING SERVER CATEGORIES

Before examining the detailed components of a server, you need to know the general categories of servers and understand how each type is used. Determining the primary use of the server helps to determine the types of components you need. For example, a file server requires high-speed disk drives, whereas application servers require high-speed processors. Determining the necessary types of components can be difficult in a Web server environment because you must consider the need for a Web server, database management servers, and e-mail servers, as well as the server requirements of programming languages and other systems.

Understanding File Servers

As its name suggests, a file server sends and receives files. For file servers, a fast disk subsystem is more important than the processor type. Nevertheless, you should make sure that the file server's processor is powerful enough to run applications efficiently.

In the Web environment, you can use a Web server that primarily contains static HTML files as a file server. Because the Web server simply sends files from its hard disk to the network, the processor does not have to do much work. However, many Web sites mix static HTML files and dynamic files that require processing by a programming language. To offset the processing burden of a programming language, large Web sites use application servers that specialize in creating dynamic pages.

You can also use an FTP server in the Web environment to transfer files, usually from the server to users. For example, when you download applications, service packs, and other large files, you are probably transferring them from an FTP server.

Understanding Application Servers

The tasks performed by an application server are more complex than those carried out by a file server. An application server runs server applications that wait in the background, ready to process requests, rather than user applications such as Microsoft Word. Typically, a server application processes requests from many users at the same time. For example, a server that contains a database management system (DBMS) is an application server. Although the disk subsystem is important, a DBMS requires extensive processing power because it often processes complex requests from many users.

An e-mail server is another example of an application server. While some e-mail servers simply transfer files that contain e-mail messages, many also process data by verifying that a user is valid and by testing connections with other e-mail servers to send and receive files. Microsoft's e-mail server, Exchange, is considered a groupware server that performs services such as collaboration, task management, and meeting management, making it an application server.

A Web server that sends static HTML files to the network requires no logical operations or file processing. However, when the Web server adds support for programming languages such as Active Server Pages (ASP) or JavaServer Pages (JSP) technology, the HTML pages work like applications. For example,ASP might process a text file to search a database and produce a report on certain products. These operations could require extensive processing power. When considering which servers to select, you need to determine how they will be used, which applications will run on them, and how the applications will be used. To help select the appropriate mix of features, also consider whether these applications are more disk intensive or processor intensive.

EVALUATING the COMPONENTS of COMPUTER PERFORMANCE

As you learned in the previous section, the purpose of the server determines which components you need. Just as you need to balance the parts of a Web server system, so you also need to balance components within the server computer. This requires an understanding of your core processing needs to appropriately size the CPU and Memory, the Disk Subsystem and the Network Connection. An inappropriately sized system will have performance **bottlenecks**. Ideally, all the components in a server should work together to optimize performance.

Evaluating Processors and Memory

The processor is the main item to consider when purchasing a server because it makes everything else work. If the processor is a bottleneck, the solution may involve an expensive upgrade.

Examining the Intel Family of Processors

Most Microsoft Windows computers use Intel processors. The most common type of Intel processor sold today has a 64-bit chip. When the processor chip size doubled from 32 to 64, it effectively became more than twice as fast. This is in part because the addressable memory went from 2^{32} (=4 gigabytes) to 2^{64}

(=16 exbibytes). https://en.wikipedia.org/wiki/64-bit_computing Remember, the CPU and memory are tightly coupled. A CPU fetches and executes instructions and data in its registers that are memory resident. The CPU's performance is very dependent on memory access speed which uses the **system bus** http://en.wikipedia.org/wiki/Bus_(computing). A specialized form of high-speed memory know as **cache** is built directly into the CPU to enhance performance. Level 1 (L1) cache is first accessed followed by L2 and so on. http://en.wikipedia.org/wiki/L2_cache#MULTILEVEL. **Bus speed** in general is the rate at which signals are sent between devices such as memory, the hard drive and network interface card. Data from peripherals (Disk Subsystem and Network Connections) must first be moved to memory at a much slower before it is CPU accessible. Look at data transfer rates here: https://en.wikipedia.org/wiki/Hard_drive & https://en.wikipedia.org/wiki/Network_interface_controller Generally speaking, increasing memory is the least expensive way to increase server performance.

Intel is the dominant supplier of microcomputer processors although they don't currently dominate mobile device market. Their 64 bit client offerings include the Core i series and server offerings include Pentium Xeons. https://en.wikipedia.org/wiki/List_of_Intel_microprocessors

Using Multiple Processors

One way to prevent the processor from becoming a bottleneck is to use more than one processor and spread the work among the various processors. Some applications, such as BEA WebLogic, can assign several servers to a single processor. Other applications, such as a DBMS, are designed to work together across processors. Other applications cannot benefit from multiple processors at all. A server's ability to use multiple processors depends on the combined capabilities of the processor, the motherboard, the operating system, and the design of the application. With Intel processors, the Xeon is used in servers with multiple processors.

Evaluating Hard Drives

After the processor and memory, the disk subsystem may be the most important server component. It includes two parts:

- The hard drive interface, which connects drives to the motherboard
- The hard drive itself

The hard drive can significantly affect overall system performance. When choosing a hard drive interface, balance the desired performance with your budget. **Serial ATA** is an inexpensive alternative for connecting host bus adapters to mass storage devices. For servers that need higher performance and scalability, the **Small Computer System Interface (SCSI)** interface is a better choice.

Using Serial ATA Interface

Serial ATA (SATA) is a computer bus interface that connects host bus adapters to mass storage devices such as hard disk drives, solid state drives and optical drives. Serial ATA replaces the older AT Attachment standard offering several advantages over the older interface including: reduced cable size and cost, faster data transfer. SATA host adapters and devices communicate via a high-speed serial cable over two pairs of conductors. SATA drives are limited in terms of expandability and overall speed. https://en.wikipedia.org/wiki/SATA

Using the Serial Attached Small Computer System Interface

Serial Attached SCSI (SAS) is a point-to-point serial protocol that moves data to and from computer storage devices such as hard drives and tape drives. SAS replaces the older Parallel SCSI (Small Computer System Interface, pronounced "scuzzy") bus technology that first appeared in the mid-1980s. SAS, like its predecessor, uses the standard SCSI command set. SAS offers backward compatibility with

SATA, versions 2 and later. This allows for SATA drives to be connected to SAS backplanes.
https://en.wikipedia.org/wiki/Serial_Attached_SCSI

Selecting a Hard Drive

A hard disk drive (HDD) is a data storage device used for storing and retrieving digital information using rapidly rotating disks (platters) coated with magnetic material. An HDD retains its data even when powered off. Data is read in a random-access manner, meaning individual blocks of data can be stored or retrieved in any order rather than sequentially. An HDD consists of one or more rigid ("hard") rapidly rotating disks (platters) with magnetic heads arranged on a moving actuator arm to read and write data to the surfaces.

A key measure of hard drive performance is the **access time**, or the amount of time it takes the drive to retrieve a single piece of data. The access time, which is measured in milliseconds, includes the seek time, or time needed for the drive's read/write head to find a particular cylinder on the disk. The seek time is typically higher for a larger disk because it has more space to search. Another factor that affects access time is the spindle rotation speed of the drive, also referred to as the **drive speed**. A higher rotation speed lowers the access time. When evaluating hard drive performance, consider the following factors:

- *Vendor*—You should select products from a reliable vendor. Investigate the **mean time between failure (MTBF)**, which is the average time interval that elapses before a hardware component fails and requires service. Also find out what kind of support the vendor provides.
- *Capacity*—You can choose from a wide range of capacities, typically starting at about 9 GB and increasing significantly.
- *Data transfer rate*—This rate can be represented by two speeds—hard drive to buffer and buffer to adapter.
- *Buffer size*—The buffer consists of RAM storage between the adapter and the hard drive. Buffer size is measured in megabytes.
- *Average seek time*—This measure indicates the time it takes, in milliseconds, to get to a position on the drive.
- *Rotational speed*—This measures how fast the disk drive spins. Typical rotational speeds are 7,200 RPM, 10,000 RPM, and 15,000 RPM and higher.

You can configure multiple drives in a system in many ways. SCSI adapters can have multiple drives to expand the system's storage capacity. This approach can also improve the system's overall speed if you divide the drives to isolate the operating system on one drive and the applications on the rest of the drives. Isolating system components is a relatively inexpensive solution, but each drive introduces another potential point of system failure. That is, if one drive fails, the whole system fails. To circumvent this problem, you can use a **redundant array of inexpensive/independent disks (RAID)**, a common drive configuration on servers. RAID allows multiple drives to operate together as a single drive, and it often uses a SCSI interface. If one drive malfunctions, the system continues to work. This stability is part of the important concept of fault tolerance which is discussed later in this chapter.

In Hands-On Project 2.3 you will discover how Solid State Drives have dramatically changed disk storage technology.
https://en.wikipedia.org/wiki/Hard_disk_drive_performance_characteristics
https://en.wikipedia.org/wiki/Solid-state_drive

Evaluating a Network Interface Card

The network interface card (NIC) is another server component that can affect overall performance. The NIC provides the pathway for data to enter and leave the server. Table 2-3 provides an overview of common NIC types and speeds along with their usage. Notice that 100 Mbps & 1 Gbps are commonly used NIC speeds over copper.

Table 2-3 Common network interface cards

NIC Type	Speed	Media	Use
Standard Ethernet	10 Mbps	Twisted Pair (sometimes fiber)	Workstations
Fast Ethernet	100 Mbps	Twisted Pair (sometimes fiber)	Workstations and small to medium-sized servers (most popular)
Gigabit Ethernet	1 Gbps	Fiber (sometimes twisted pair)	Workstations and small to high-end servers
10 Gigabit Ethernet	10 Gbps	Fiber	High-end servers

You can connect two NICs to a switch and configure the NICs so that if one fails, the other will continue to supply data. This approach provides an inexpensive kind of insurance against NIC failure. In a server environment, redundancy is used to prevent a single point of failure.
https://en.wikipedia.org/wiki/Network_interface_controller

It is usually best to choose NICs from major vendors.
http://en.wikipedia.org/wiki/List_of_networking_hardware_vendors#Network_interface_card

Comparison of LATENCIES for each COMPUTER COMPONENT

1 CPU cycle	0.3 ns	1 s
Level 1 cache access	0.9 ns	3 s
Level 2 cache access	2.8 ns	9 s
Level 3 cache access	12.9 ns	43 s
Main memory access	120 ns	6 min
Solid-state disk I/O	50-150 µs	2-6 days
Rotational disk I/O	1-10 ms	1-12 months
Internet: SF to NYC	40 ms	4 years
Internet: SF to UK	81 ms	8 years
Internet: SF to Australia	183 ms	19 years
OS virtualization reboot	4 s	423 years
SCSI command time-out	30 s	3000 years
Hardware virtualization reboot	40 s	4000 years
Physical system reboot	5 m	32 millenia

http://blog.codinghorror.com/the-infinite-space-between-words/

Planning for System Disasters

Disaster planning can help you avoid problems with hardware, software, and even business procedures. Planning for system disasters is like buying insurance—you might not like paying for it, but you are glad to have it when you need it. Also, just as it may be economically infeasible to insure your company against every conceivable problem, it may not make good business sense to make sure that systems will never fail. Always balance the cost of disaster planning against the benefit to the organization and others. Servers can be critical to a business. For example, on an e-commerce site, a server failure could cost a business thousands of dollars for each minute of lost revenue. Virtually anything you can do to keep servers running represents time and money well spent, but spending money on **fault tolerance** must provide a distinct business benefit. Fault tolerance is the ability of a system to keep running even when a component fails. Recall from Chapter 1 that virtualization provides high availability through clustering and can more efficient use of CPU, RAM and disk.

Not every server needs to be 99.999 percent reliable (the coveted "five nines"), which would mean about five minutes of downtime per year. This level of reliability can be expensive to achieve, and only critical servers need such high reliability. For example, DNS servers have built-in fault tolerance. (Recall that a DNS server converts host names to IP addresses for your domain.) You might have a local DNS server and your ISP might have another DNS server. If your DNS server malfunctions, the DNS server at your ISP can take over and resolve names. This fact means that your local DNS server is not a critical server. Indeed, some companies use recycled workstations running Linux as a DNS server. They keep all the DNS scripts on a floppy disk and set up an old workstation to use as a DNS server when necessary.
In creating a reliable system, you need to justify the cost involved. Top-level managers usually make the initial decisions to address the cost of business downtime, though they often do not realize its full impact. You can help them in this effort by doing a disaster assessment.

Disaster Assessment and Recovery

To understand how disasters would affect your system and business, start by identifying which disasters could strike your server and pinpointing how long they could last. Determine which disasters could result from computer malfunction, simple human error, or the actions of disgruntled employees.

Focus on the disasters you can prevent. For example, a server that uses a single disk drive might take advantage of RAID technology to prevent problems in case the drive fails. Lost data from an accounting program could be restored by a tape backup, by the program itself (which might offer an option to restore the data), or by a database administrator (who might restore the data from detailed logs kept by the DBMS).

Think creatively about disasters that can cause monetary loss and ways to prevent them. If a disaster does occur, have a recovery plan that minimizes the cost to the organization. Be sure to maintain adequate documentation of your systems. Because you are responsible for the Web server environment, you could be the first person blamed for its failure. Document the hardware, software, and configuration decisions made by you and your managers.

Preventing Hardware Disasters

The hardware problems on which Web server administrators focus are those involving the server. Because these computers are complex, you need high-quality technical support for them. However, servers aren't the only devices that can fail—all the components in a Web server environment must work together. For example, a working server won't help you if the router that connects it to the Web fails.

For this reason, you need a plan of action to address hardware failures in a Web environment. If a component fails and you have support for it, for example, you need to know the support phone number and the location of the support contract. The support person may in turn need the contract number and the serial number of the failed component. Make sure that these numbers are written down and that more than one person knows about them, because components can fail when you are on vacation.

If you do not have support on a device, a disaster recovery plan is even more important. If the router fails and you need to purchase a new one, you might need preapproval to buy it. Management needs to know in advance the cost of the component, the cost of support for the component, its role in the Web environment, and the importance of expediting an order in case the component fails.

Preventing Software Disasters

Software disasters are more complex than hardware disasters because so many types of software exist. No administrator can be expected to understand the intricacies of every application in a company. However, the administrator is often the person responsible for knowing who to call to get a problem solved. When the problem involves a computer, you are likely to be the first person whom users call for help. Software companies often have support lines that you can recommend as a resource. For a complex DBMS, your company may not need or be able to afford a full-time database administrator, so it might contract with a local firm to fix problems.

As with hardware components, you need to document every software component and devise a plan for dealing with problems. The plan could be as simple as providing a list of phone numbers for internal software experts. In other cases, you might need to call the software support number. As with hardware, a software application may offer only 30 days of free support. If a software maintenance contract is needed, you need to make management aware of the annual or per-incident cost. If you need to pay a per-incident cost in an emergency, make sure you can have the payment expedited so you can get support. Typically, your company should have a general budget for emergency support.

Web server administrators occasionally encounter software that does not work properly from the day it is installed. It may not work correctly, it may stop unexpectedly, or it may lose data. Also, it may not have features that were promised. If the software is important to the company, however, it is your job to make it work successfully.

In such situations, be sure to document your problems with the software and the actions you take to solve them. Document your conversations with support personnel, any patches you apply to the software, and any other procedures you perform to fix the problems. Beware of relying too much on workarounds. For example, a support technician may tell you that when the application stops, you can simply go to the Web server and restart the service that controls the application. This workaround may suffice when you are available, but it won't help on a weekend, when no one is near the Web server.

Solving Electrical Problems

The old saying that "the memory of a computer is only as long as its power cord" highlights the importance of a constant electrical supply. Even if the original server room was expertly planned and the electrical needs of each component were carefully researched, with dedicated circuits provided for all components that required them, electrical problems can arise later. For example, servers might be added and other components might be upgraded without the addition of any new circuits. If components are not attached to an **uninterruptible power supply (UPS)**, an overloaded circuit could cause a component to restart itself. If you overload the UPS, the battery within it will become too drained to keep components running. Make sure you have an adequate supply of electricity and enough UPSs for your server room, and make sure each UPS has enough capacity.

In a large, complex environment you need an expert to tell you how many circuits and UPSs you need. Nevertheless, you can typically make a reasonably good estimate yourself. Start by calculating **watts**, a unit of power. Wattage is equal to volts multiplied by amperes (amps); each circuit is usually 15 or 20 amps and 110 to 120 volts, so the number of watts for a 15-amp circuit is about 1,725 (115 * 15).

Your next challenge is to find out from building maintenance which wall sockets are part of which circuit. In some cases, you could have a dedicated 20-amp circuit for a single device; often, however, a number of wall sockets will be part of the same circuit. Offices adjoining the server room might also share a circuit. Once you map the sockets to the circuits and find out how many amps are on the circuit, you need to know which components will use that circuit. The components' power supplies usually indicate the number of watts they use. Typically, a power supply on a server may be 300 watts, but the environment might include three power supplies. Total the wattage and find out how many circuits will be needed.

A 300-watt power supply is not like a 300-watt bulb. The power supply does not consume a constant 300 watts, but only the power that it actually needs at a given time. However, make sure that your electrical supply can give you maximum power so that you have room to expand. Remember to gather information on future needs so your server room has enough circuits. Also, keep in mind that some devices, such as large Cisco switches, need significantly more watts when they start up than when they are running. Make sure that your supply of electricity can handle this kind of fluctuating demand.

In a server room, you should always place a UPS between the wall socket and the devices. A UPS is rated in watts, so you can calculate how much power you need. Also, consider how long the server should run on the battery when the electricity goes out. In most cases, you want just enough electricity to shut down the servers properly. Software/hardware combinations from the UPS manufacturer can handle this task automatically, so if an electrical outage occurs when no one is near the server room, the servers will shut down without any human intervention. When servers are shut down properly, data in RAM that could be critical to the operating system and applications is stored on the disk where it belongs. For example, files that are open are closed properly. If systems lose power and cannot shut down correctly, data can be lost and hard disks can suffer serious problems.

Although disaster assessment and recovery are complex tasks, Web server administrators are not solely responsible for preventing and recovering from disasters. In larger organizations, for example, the help desk is charged with resolving many application software problems. The IT director should have an adequate budget to make sure that any monetary issues related to support get resolved quickly.

Allowing for System Redundancy

After acknowledging that hardware occasionally fails, you need to assess which components are most prone to failure and then determine how to avoid failure, how much it costs to avoid it, and whether the cost justifies the benefits to the organization. This section will help you determine which components are most likely to fail and learn how you can avoid failure through redundancy. Power supplies on larger devices such as servers and switches are not only critical components, but also relatively inexpensive. Often, servers have two or three power supplies; switches may have two such supplies. If some of your servers have only one power supply, you should purchase spares.

Achieving Disk Redundancy Through RAID

A RAID system can prevent data loss when a single drive malfunctions. The RAID levels represents a different way to make multiple disk drives act as a single drive. Although RAID 0 is not fault tolerant, the other levels are.

The RAID levels are described in the following list:
▪ *RAID 0*—In this technique, called disk **striping**, data is split into small pieces and spread over a number of drives. Disk striping in RAID 0 is the fastest of all the RAID levels because it does not have to store the data necessary to allow a single disk to fail.
▪ *RAID 1*—Data is duplicated across two drives, which can make writing data slower. This method, called **mirroring**, is ideal for SATA drives because you can have only two high-speed drives. If you are using SCSI, you can have two SCSI adapters; each goes to a drive, so even if one of the adapters fails, the system will keep running. This technique is called disk duplexing. Figure 2-9 compares these approaches.

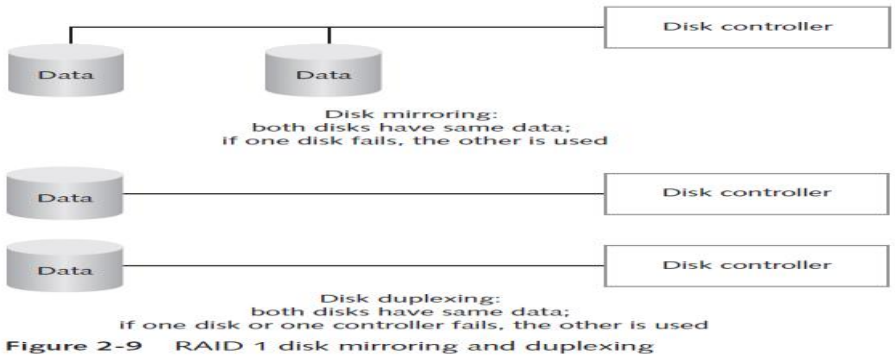


Figure 2-9 RAID 1 disk mirroring and duplexing

■ *RAID 5*—This technique is by far the most common disk redundancy method found in servers. RAID 5 distributes data across three or more drives and stores the redundancy, or parity, information needed to rebuild the drives; thus, if drive fails, it can be replaced with no loss of data. Although all Windows can implement RAID 5 in software, it requires too much extra memory and processing power to be generally feasible. It is much less of a burden on the operating system to have RAID 5 implemented in hardware via special SCSI RAID controllers. All major server vendors configure their servers in this manner.
▪ *RAID 10*—Instead of using an array of single drives, RAID 10 is an array of RAID 1 mirrored drives. It is much more expensive than the other RAID levels because you have twice as many drives. RAID 10 is better than RAID 5 because it eliminates the immediate need to replace a drive and each disk in the array is mirrored. Also, it is more fault tolerant than RAID 5; if two disks in a RAID 5 array fail before one is replaced and the array is rebuilt, then you will lose your data. With RAID 10, two pairs of disks would have to stop functioning before you had to replace any drives.

Disk mirroring and disk duplexing (RAID 1) are very common in the low-priced segment of the server market. Setting up disk mirroring in a Windows server with IDE drives is very easy. If your computer has two 20 GB drives, you can install a Windows server on one of the drives and make it a dynamic disk (as opposed to a basic disk). In the Disk Management utility, you can mark the other disk and create a mirror. If one of the disks fails, you can then use the Disk Management utility to break the mirror, which makes the disks independent again. At that point, you can simply replace the broken drive and re-create the mirror.

Outside of the low-end market, RAID 5 is the most popular method of disk redundancy. In mirroring, you can have only two drives; in RAID 5, however, you can have dozens of drives. The equivalent of one drive is used for redundancy and you need a minimum of three drives. For example, if a RAID 5 array includes three 20 GB drives, the storage capacity would be 40 GB. With ten 20 GB drives, you would have 180 GB of storage. If you have *n* number of drives, with *n* being at least 3, and *y* is the capacity of each drive, the available storage capacity is (*n* - 1) _ *y*. If one of the drives then malfunctions, the other drives have the data from the malfunctioning drive, so the only difference is a slight slowdown because data must be re-created from the other drives. Once you put a new drive back into the system, it is automatically rebuilt. Some RAID 5 installations allow for a hot swap meaning that you can replace the drive while the server continues running.

All major server vendors support hardware-based RAID 5 systems. Most RAID 5 technology is implemented in hardware through a special utility in the RAID 5 controller. Although Windows server products allow you to set up RAID 5 in software, you should avoid using software-based RAID 5 because of the processing and memory burden it imposes on the operating system.

A Storage area network (**SAN**) is a dedicated network that provides access to consolidated, block level data storage. SANs are primarily used to enhance storage devices, such as disk arrays accessible to server so that the devices appear like locally attached to the operating system.
https://en.wikipedia.org/wiki/Storage_Area_Network

Achieving High Availability with Multiple Servers

Clustering is a technology in which many computers act as one. You can also use a simpler technique called load balancing to distribute the work among many computers.

Clustering has three major purposes:

- *Computing power*—You may need so much computing power that no single computer can handle the demand. A cluster of relatively inexpensive computers can offer more computing power than one large supercomputer. These types of servers are often used in the scientific community to handle complex applications.

- *Fault tolerance*— At the other end of the spectrum from computing power is fault tolerance. Pure fault tolerance is very difficult to achieve, because clusters are very complex and includes many components all of which can be points of failure.
- *High availability*—Between the two extremes lies high availability through clustering. This approach provides redundancy and failover for fault tolerance, but its definition is not quite as strict as that of pure fault tolerance. (Failover is the ability to have a server fail and yet have the other servers continue to function.).

IBM's approach to clustering is more complex than Microsoft's approach. IBM uses computers it calls nodes to serve different purposes. Most servers are used as compute nodes, which is where the real work occurs. In contrast, Microsoft's approach to clustering is to distribute the computing load (load balancing) among distinct servers. This approach to high availability must be able to accommodate any computer that uses Microsoft Windows server products. The Microsoft solution uses a software product that ties together multiple servers into a single system, and it focuses on Web server availability.

The simplest form of load balancing was a **DNS round-robin**. In a DNS round-robin, one host is associated with multiple IP addresses corresponding to multiple Web servers. As requests for Web pages arrive, they are sent to the next IP address on the list. To add new Web servers, you simply add IP addresses to the list. Microsoft uses Network Load Balancing to balance network traffic across multiple network cards.

Setting Up Backup Systems

No system can achieve fault tolerance without having a good backup system in place. A backup of your data files protects against user error, and it can prove invaluable if a data error goes unnoticed for days. Not only should you keep data backups, but you should also maintain a number of backups made over a period of time so that data can be restored from a specific date.

When setting up a backup system, consider the following issues:

▪ *Your backup procedures*—How many backup procedures do you need? How often do you need to make the backups?
▪ *The backup technologies you can use*—How many tape drives do you need in a multiserver environment? How do you make sure everything is backed up, including specialized applications such as DBMS and e-mail servers?

Backup Procedures

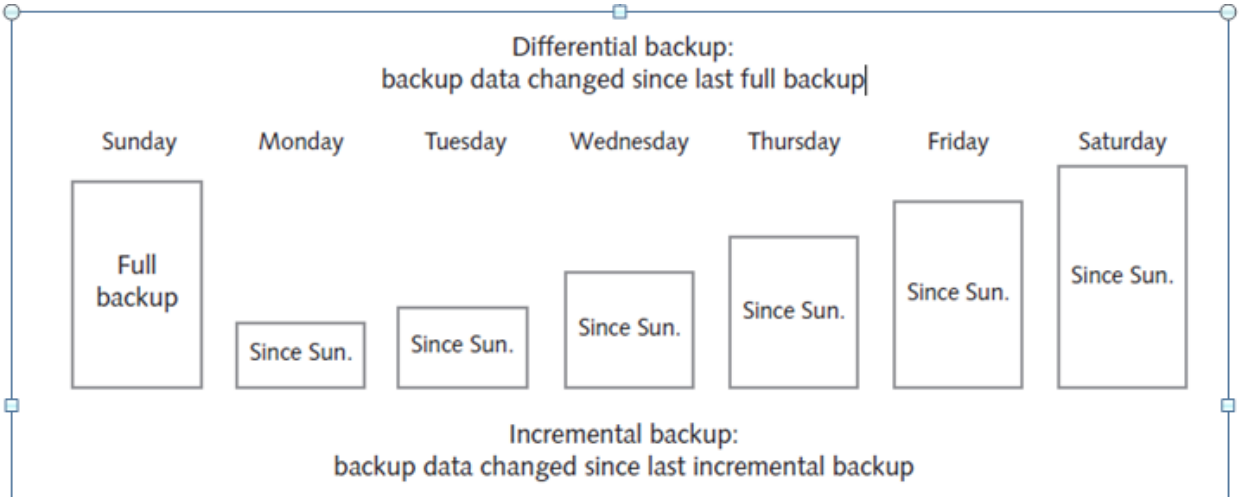
The number of tape backups you make and the frequency of backups depend on the needs of the organization. Typically, you will need daily backups for the previous week and weekly backups for the previous month. Many organizations also may have special backup needs based on certain occurrences. For example, a company may need a backup immediately before month-end closings in the accounting system and another backup immediately after the month-end closing. You may want to back up the system before installing any new software, in case the software causes serious system problems that you cannot solve by simply uninstalling it.

Once you have decided on your schedule of backups, you have to decide on the backup method. Each type entails certain trade-offs:

▪ *Full backup*—All backup schedules begin with a full backup of everything on the drive. The advantage is that all the data is in one place; if data must be restored, you need to access only one storage location. The disadvantage is that a full backup takes more time than other techniques. You will use this technique to backups your VMs at the end of each chapter.

▪ *Differential backup*—This method backs up all files that are new or changed since the last full backup. The advantage is that if data must be restored, it can be found in only two possible places: on the tape with the full backup or on the tape with the differential backup. The disadvantage is that you keep backing up the same old information since the full backup, because you back up new data every day. If you have VM Workstation, you may elect to use Snap shots which are a differential backup variant. Essentially you'll snapshot your work when you successfully complete each chapter to provide a backup point before continuing.

The following figure shows the differential backup approach.



Often, full backups are scheduled to take place once per week during a slow time, such as 1 A.M. on Sunday. During the week, incremental or differential backups might be done at 1 A.M every morning. Store the weekly full backups off site, in case the computer room becomes damaged by fire or fire sprinklers.

Choosing Backup Technologies

You can back up data in more than one way and should consider whether you need more than the basic procedures with specialized consideration for multiple LAN servers, multiple Web servers, a database server, an e-mail server, etc..

You should consider three issues when performing backups:

- **Backing up the operating system**. In a Windows server, the Registry is always open, and open files are not backed up by default. You need to make sure that your backup software explicitly backs up the Registry. **Windows Server 2012** has its own specialized backup procedure.
- **Backing up special application files**. DBMSs keep files open, as do Microsoft Exchange and other applications. These critical files will not be backed up unless your backup software has special modules. Often the only way to close the application files may be to manually stop the application before you make a backup. Some applications, such as DBMSs require part of the backup procedures to take place through utilities that are part of the DBMS software.
- **Backing up simple data files such as text files, spreadsheet files or executable files**. These files are easy to back up, assuming that the user does not have them open. Make sure you can back up all files to ensure that you can rebuild the system after a disaster. This may not be a trivial task. Verify that your software can restore the system by practicing on every new server you get and documenting the procedures.

Backup systems can require a significant amount of network overhead. To prevent this problem, backups should always be done when no one is using the network. In addition, put an extra NIC in each server and implement a second network just for backups.

EVALUATING NETWORK COMPONENTS

There is more to a Web server environment than a few servers. The servers need to be connected together and the connections need to communicate with the Internet. This section examines these components and explains how to put together a complete system.

Switches and Hubs

You use switches and hubs to connect computers. A twisted-pair wire usually connects the NIC in the computer to the hub or switch. A switch is a device that controls the routing and operation of data signals. Standard switches communicate at Layer 2 of the OSI model. However, as explained in Chapter 1, Layer 3 switches can also act as routers. Hubs are shared devices found at Layer 1 of the OSI model; computers share the connections in the hub much like old-fashioned telephone party lines. Only one conversation could take place at a time. In the computer world, this situation is called **contention**. The more traffic, the slower it travels. Using a hub between a workstation and a server is fine with light data traffic, but switches are a much more common solution to heavier traffic management.

A switch is analogous to a telephone switch. You dial a specific number and then communicate, even though others may be using the same telephone system at the time. A switch simulates a direct connection between two computers. Although these devices should be really called switching hubs, they are commonly called switches. Because servers handle a lot of traffic, this section focuses on switches; hubs are not an option when connecting servers.

Not all switches are the same. As with most components, you need to balance your needs with your budget. For example, assume that you have a 12-port switch. (A port is where you connect the network cable.) When only two computers are connected to a 12-port switch, traffic flows without interruption. As traffic increases, however, you need to consider the following characteristics:

- *Packets per second*—The number of packets that can go from one port to another port. More expensive switches approach **wire speed**, which is the same speed two computers could achieve if they were physically connected.
- *Data switching backplane*—This is the total speed the switch can handle. It should be measured in gigabits per second.
- *Connection types*—Ask questions: Can you use full-duplex NICs that allow 100 Mbps data transfers in both directions at once? As you add more switch capacity, can the switches function as if they were one switch?

Once you have a switch to connect the servers in a network, you must connect the server network to your ISP's network. Your ISP then connects your system to the Internet.

Routers

Routers connect one network to another network and can serve many purposes, including connecting an internal network to an external network. Chapter 1 discussed connecting your network to the Internet. Recall that the digital signal coming from the Internet differs from the digital signal in your network. The router not only moves packets from one network to another, but can also transform the packet into another type. For example, when a router links to your internal network and your CSU/DSU, its Ethernet port connects to your internal network and its serial port connects to your CSU/DSU.

Not all routers are separate devices, and any server can become a router. All you need are two or more NICs. The connection into one NIC comes from one network, and the other connection goes to the second network. A firewall computer can take packets from the Internet on one NIC and then send them to an internal network with a special network address that cannot be detected from the Internet on the other NIC. A firewall computer can also act as a router.

Maintaining Internet Connections

There are many pieces to the puzzle of identifying LAN components and connecting them to the WAN, as you discovered in Chapter 1. The most complex piece involves learning about the WAN connection. Due to the competitive nature of the ISP industry, both pricing and services change rapidly. As a consequence, the most popular and cost-effective solution one year may not be the best solution the next year. Although the T-Carrier approach offers virtually unlimited expandability, many businesses may not need the expandability. They can select between T1, DSL, and a cable modem. The choice will depend on the combination of services that are available in your area, the cost of the service, the reputation of the ISP, and the expandability you need.

Organizations that are just getting started in Internet connectivity should consider the Web hosting solutions outlined in Chapter 1. They involve much less risk and do not need on-site technical expertise.

SETTING UP IP ADDRESSING

As you learned in Chapter 1, IP is one of the protocols in the TCP/IP protocol suite. Its purpose is to provide addressing, which is how information gets from one computer to another on the Internet. Every Web server has a unique address that is valid on the Internet. However, workstations on an organization's LAN often use private IP addresses that are not accessible on the Internet, but rather have meaning only on the company network.

Understanding the Addressing Structure

The addressing structure determines how addresses are created and how you can determine the difference between the **network** portion of the address and the **host** (or individual computer) portion.

IP addresses are divided into four numbers separated by periods, such as 192.168.0.100. Each number, with certain restrictions, can range from 0 to 255. An IP address has two parts: a host portion and a network portion. All computers that are directly connected to each other form a network with regard to IP addressing. In such a case, the network portion of the IP address must be the same for all computers, and the host portion of each machine's address must be different to distinguish one computer from the rest of the computers on the same network.

Subnet Mask

The subnet mask tells you what part of the IP address represents a network number and what part of the address represents the number for the host. IP addresses are classified into three principal classes, as shown in Table 2-4. The addresses are grouped according to how many hosts each class can accommodate. A **class A** address can have over 16 million hosts, whereas a **class C** address can have only 254 hosts. Figure 2-13 gives an example that shows the differences among the classes with regard to network and host portions of the address.

Table 2-4 Common TCP/IP classes

Class	First number	Subnet mask	Number of networks	Number of hosts
Class A	1–127	255.0.0.0	126	> 16,000,000
Class B	128–191	255.255.0.0	> 16,000	> 65,000
Class C	192–223	255.255.255.0	> 2,000,000	254

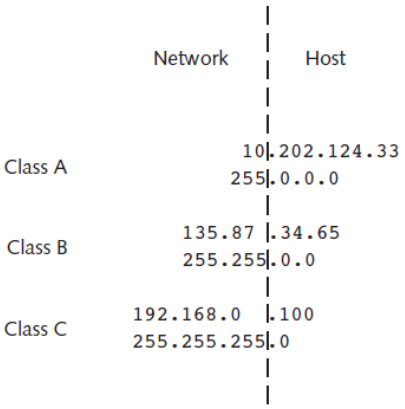


Figure 2-13 Subnet mask used to separate network from host

Determining the network portion and the host portion is critical for the following reason: when your computer needs to send a packet based on an IP address, it must determine whether the packet should stay on the local network or be sent through the gateway to another network. To do so, your computer compares the IP address of the destination to the subnet mask. If the network portions of both your computer's address and the destination address match, the packet stays on the local network. If the network portions are different, the packet is sent to the gateway (router) address. When you set up an IP address in your computer, the third value—usually called the **gateway**—is the IP address of the computer that will take the packet out of the local network so that it can ultimately be routed to the correct network.

Private Networks

Private networks are special network addresses reserved exclusively for use on networks that do not communicate across the Internet. These networks offer two advantages. First, you don't have to worry about packets from the private network getting to the Internet, because Internet routers cannot route packets that use these addresses. Second, hackers cannot easily access computers in your local network that use such IP addresses.

The private addresses have the following designations:

- 10.0.0.0–10.255.255.254 (a single class A network address)
- 172.16.0.0 –172.31.0.0 (16 class B network addresses)
- 192.168.0.0–192.168.255.254 (256 class C network addresses)

Your objective, whether for your Web server or for users in your organization, is to achieve interaction with the Internet. Private network addresses become very powerful in this effort when they are combined with **network address translation (NAT)**.

Network Address Translation

NAT allows an IP address from one network to be translated into another address on an internal network. You need to use NAT if your ISP gave you only one address for your organization instead of 254 addresses for your servers and users. Some routers and firewalls allow you to take single (or multiple) IP addresses that are destined for your network from the Internet and translate them into your local set of addresses. This approach allows you to have a single IP address of 38.246.165.10, for example, which is then translated to the address of your Web server at 192.168.0.100. Although this technique does a good job of isolating your Web server, NAT can do even more. It can take a single IP address that is valid on the Internet and translate it into a pool of local addresses. For example, 38.246.165.10 may be translated into a pool of addresses ranging from 192.168.3.1 to 192.168.3.254. Now as many as 254 users can share a single Internet connection. Figure 2-14 shows an example of this type of network. This technique has been very useful in allowing a dwindling pool of valid Internet IP addresses to serve an ever-increasing number of Internet users. Also, by ensuring that your internal IP address pool is a private network address, you make it more difficult for a hacker to penetrate your system.

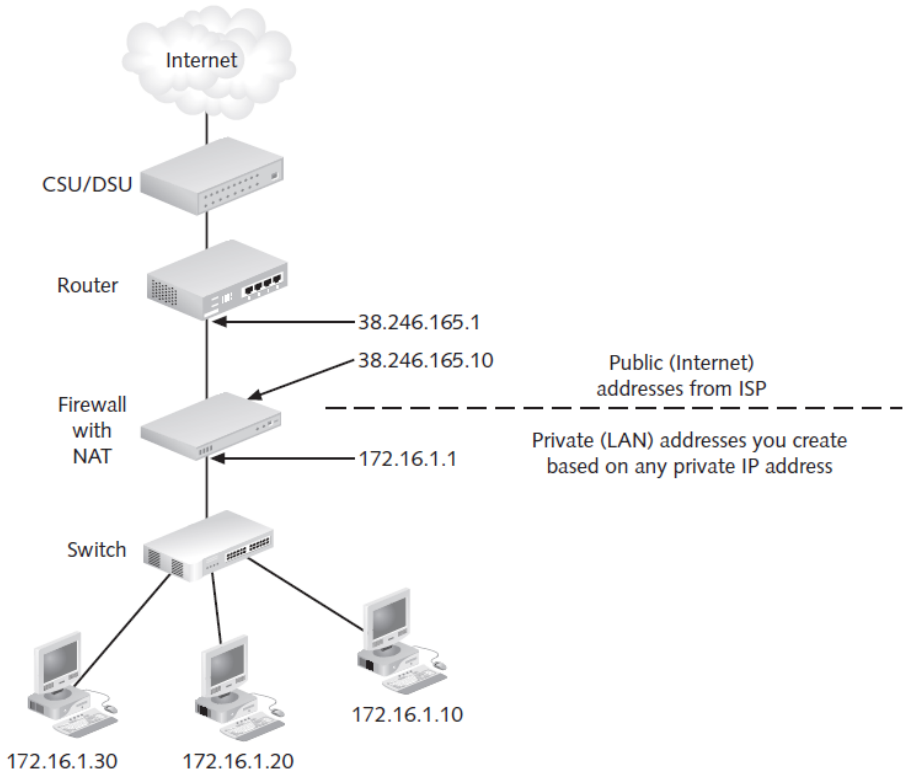


Figure 2-14 Network diagram

NAT is very flexible. For example, you could receive three IP addresses from your ISP: one destined for your Web server, one destined for your e-mail server, and one destined for your FTP server. All three will be translated to different internal IP addresses to help protect your servers.

An important aspect of NAT is that it allows multiple internal users to use a single IP address on the Internet. This type of single-address NAT is called port address translation (PAT).When a browser connects to a Web site, it typically links to port 80 on the Web server. However, for the Web server to send the Web page back to the browser, it needs to access a specific port on the browser. This port information is sent to the Web server when the user initially requests the Web page. Then a device that uses PAT, such as a router, associates each internal user with a different port. When a Web page comes back to the assigned port on the router, the port is translated into the user's port and the Web page is sent to the user. See Figure 2-15.

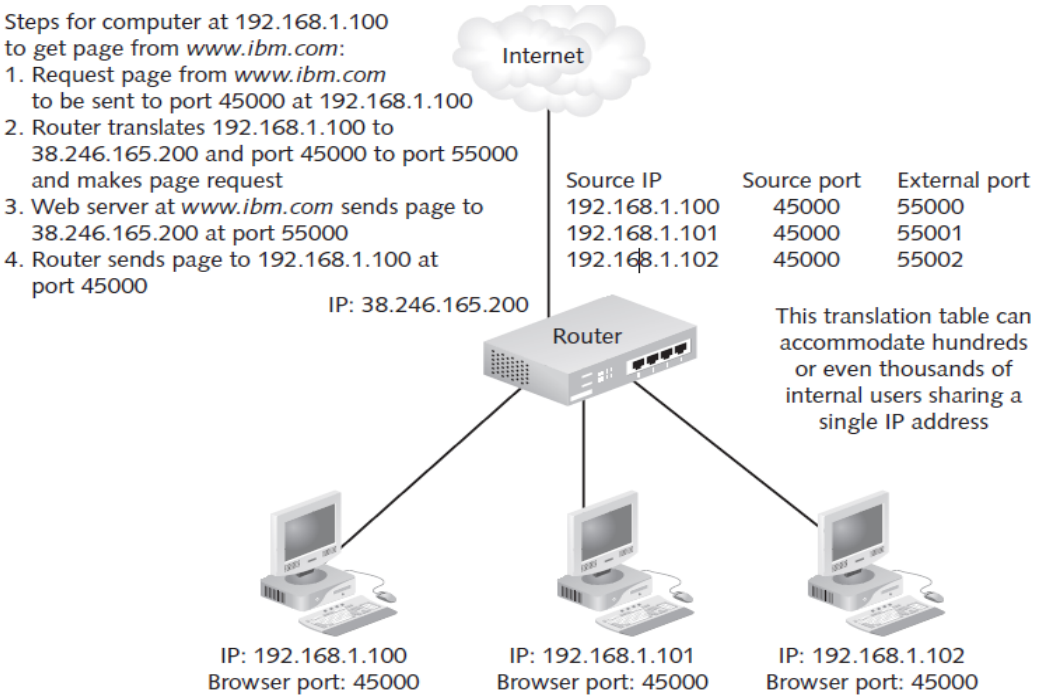
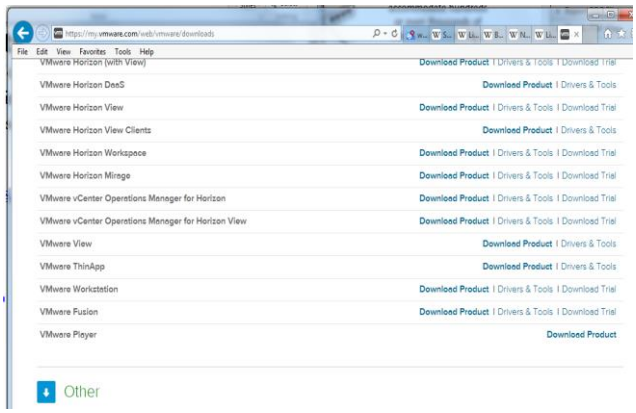


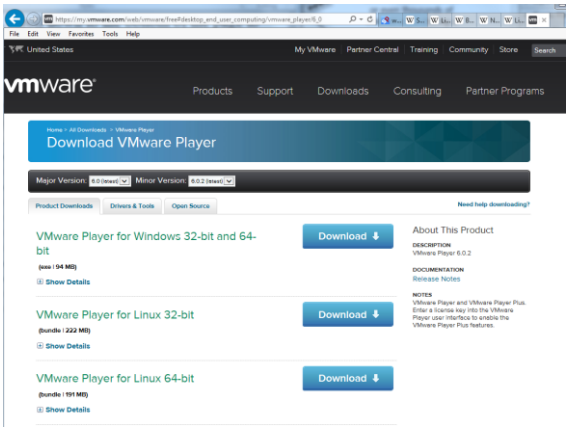
Figure 2-15 Using port address translation

Activity 2-1. Install VMware Player

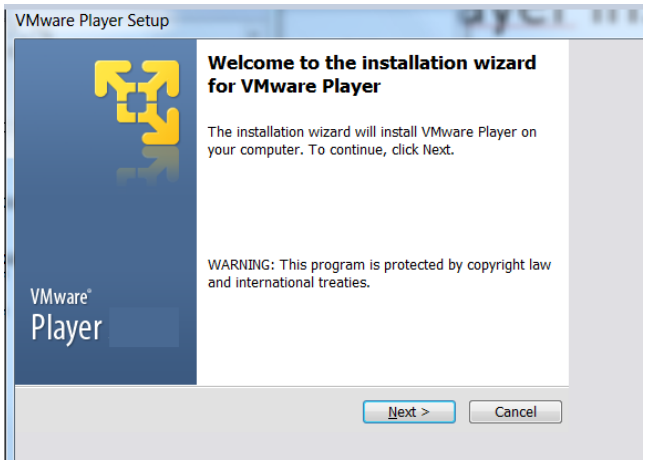
This activity will step you through a VMware Player installation. You are welcome to use VMware Workstation if you like. It has additional functionality including snapshots but you will have to get a license key from VMware. At the time this textbook was revised, the current version of VMware Player was 6.0.

1. Go to https://my.vmware.com/web/vmware/downloads, scroll down to the bottom of the page to find VMware Player and select 'Download Product'.





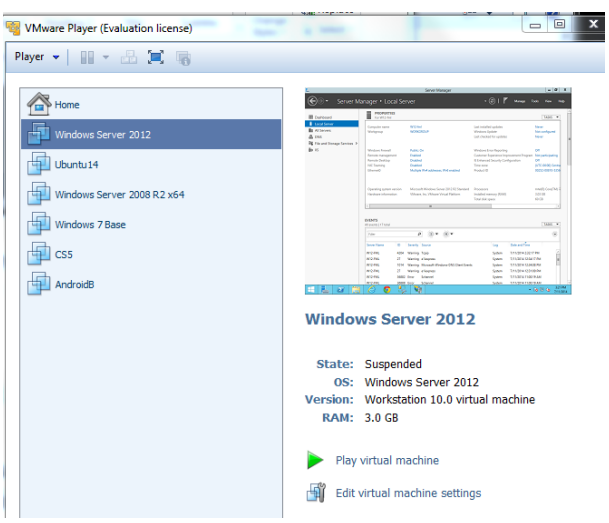
2. Next select VMware Player for Windows 32 & 64 and save VMware-player-6.0.2-1744117 in a convenient place in your download folder
3. Run VMware-player-6.0.2-1744117 and accept defaults **and take a screenshot**

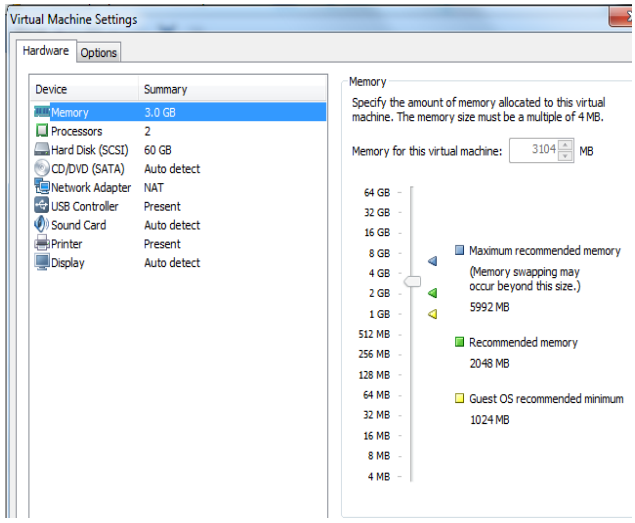


After installing VMware Player (next activity) and installing the Virtual Operating Systems (done in Chapter 3), you'll have the opportunity to fine tune your Virtual Machine Settings including the Processor and Memory resources and CD/DVD device. See screenshot below:

- From the VMware Player Start Screen you'll choose Edit Virtual Settings. (You typically shut down your machine before changing Virtual Machine Settings because you don't have access to all devices from a suspended virtual OS.)

- Your devices are listed in the left pane. After monitoring Host & VM performance, you'll have an opportunity to fine tune the Memory and Processor resources of your VM. You'll use the CD/DVD device to point to .iso for software installations. Notice that VMware Player makes recommendations in the right pane.





Be sure that VMware Tools are installed when you do your OS installations in Chapter 3 because it provides critical functionality including

- Improved graphic performance
- Improved mouse integration
- The ability to cut and paste between the host and virtual environments.

CHAPTER SUMMARY

❒ The two basic types of servers are the file server and the application server. Often, it is not easy to distinguish between them. For example, a Web server could be considered a file server or an application server, depending on how you use it. How you categorize a server affects the capabilities of the server components you choose.

❒ Many components make up a server and all of them work together to produce the appropriate throughput. The four major dimensions of performance are CPU, Memory, Disk and Network. If one of the components is not sufficient for the task, a bottleneck occurs and the server as a whole is affected. The two main Server Operating Systems are Windows and Linux.

❒ Computer components can fail and data can be lost. You must anticipate as many problems as possible and then determine how to avoid them, or at least lessen their repercussions. You can prevent many problems by providing fault tolerance or at least high availability of components in the Web environment.

❒ RAID technology is an excellent method of preventing a single disk failure from causing a loss of data. You need a minimum of three hard disks to implement RAID 5. The storage equivalent of one disk is used to provide redundancy.

❒ Clustering can achieve fault tolerance by configuring multiple servers to act as one. There are two basic types of clustering. In one approach, the cluster appears as a single computer. In the other approach, multiple servers work together.

❒ Be careful when you back up data to make sure that you include all of it; by default, open files are not copied in a backup. Many important applications, such as e-mail and DBMSs, keep files open and so have special backup needs.

❒ A complete Web server environment includes switches and hubs to connect the computers, routers to connect the networks, and Internet connections.

❒ Correct IP addressing is essential to network communication. IP addresses include both a network portion and a host portion and are classified into three categories based on the numbers of networks and hosts they can support. Network address translation (NAT) can translate a single IP address into multiple addresses that exist in the internal network.

❒ Virtual Machine Settings permit configuration of virtualized devices including CPU and Memory. VMware Tools provide functionality between the guest and host systems including cut & paste.

Hands On Projects

2.1 Create a spec sheet for the computer you intend to use as a student for your CIS studies. Include two columns headings: 'Most Economical' and 'The Best'. Include four row headings: CPU, RAM, Disk & NIC. Included in each cell are your component selections and a brief explanation using performance & cost figures of why they are 'Most Economical' or 'The Best'. Consider your CIS studies workload to determine your resource needs when doing this project.

2.2 Determine a backup strategy that you will use with your VMs in this class. In Ch 4 you setup DNS. In Ch 6 you setup a Web Servers. In Ch 7 you install Data Management Systems and write Dynamic Web Pages. What is a good plan for backing up your Virtual Machines? If you're using VMware Workstation, you can take snapshots. Define snapshots and indicate describe how they could be used in a backup scenario.

http://www.techrepublic.com/blog/virtualization-coach/how-do-snapshots-work-in-vmware-workstation/

2.3 Using these Wikipedia links, compare the data transfer rate performance of a Solid State Disk (SSD) to a Hard Drive Disks (HHD). Discuss other differences in performance including startup time, random access time, read latency time and power consumption. Why don't you defrag SSDs? - 4 points

https://en.wikipedia.org/wiki/Solid-state_drive

2.4 Use this Wikipedia link to update Table below for the following processors: Intel Core i7, Intel Core i5 and Intel Core i3. Define each of the column headings.

http://en.wikipedia.org/wiki/Comparison_of_Intel_processors

Processor	CPU Clock Rate	Number of Cores	L2 & L3 cache	Bus speed
Intel Core i7				
Intel Core i5				
Intel Core i3				

Go to D2L **Discussions** to respond to Chapter 2 Technical Tips.

Go to D2L **Quizzes** to complete the Ch 2 Review Questions. You are allowed three attempts at the chapter review questions. Your score will be the average of all three attempts.