IBM

AIX 5L and Windows 2000: Side by Side

Discover new features in AIX 5L and Windows 2000

Exploring fundamental cutting-edge technologies

Learn differencies and similarities between AIX 5L and Windows 2000

Anders Ahl
KyungHa Kim
Matt Robbins
SeungLul Son
KyeongWon Jeong

Redbooks

ibm.com/redbooks

SG24-4784-02

International Technical Support Organization

AIX 5L and Windows 2000: Side by Side

June 2001

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix A, “Special notices” on page 517.

Third Edition (June 2001)

This edition applies to IBM RS/6000 systems using the AIX 5L Operating System Version 5.0 and for PCs using the Windows 2000 Operating System and is based on information available in Feburary 2001.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. JN9B Building 003 Internal Zip 2834
11400 Burnet Road
Austin, Texas 78758-3493

Contents

Figures

Tables

Preface

The object of this redbook is to demonstrate the AIX 5L and Windows 2000 platforms to show the reader similarities and differences between each operating system. Whether you are a Windows expert looking to learn more about the latest version of AIX, AIX 5L, or are an AIX expert and are looking to inform yourself of the latest Windows platform, Windows 2000, you will find each chapter in this redbook covers the fundamental technologies that make each operating system what it is.

In ensuing chapters, we will discuss fundamental operating system concepts, architectures, open standards compliances, and product packaging for both AIX 5L and Windows 2000. Then, we shall go into the user interfaces for both, storage management, security standards compliance and operations, and full systems management. Finally, we will give an in depth discussion of networking concepts on both platforms and demonstrate the full extent of scalability and high availability on both AIX 5L and Windows 2000.

Furthermore, while not much has changed in Windows 2000, AIX is relatively new at the time of writing this redbook, and we shall make a special point of pointing out the differences between AIX 5L and the previous version, AIX Version 4.3.3.

This redbook is a minor revision from the previous version of redbook *AIX V4.3 and Windows 2000, Side by Side*, SG24-4784-01.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization Austin Center (ITSO).

KyeongWon Jeong is a Senior IT Specialist at the International Technical Support Organization, Austin Center. He writes extensively on AIX and education materials. Before joining the ITSO, he worked in IBM Global Learning Services in Korea as a Senior Education Specialist and was a class manager of all AIX classes for the customers and interns. He has many years of teaching and development experience.

Anders Ahl is an Advisory IT Specialist for IBM Global Services in Sweden. He is a Tivoli IT Director Certified and an MCSE with over nine years of experience in the Windows NT/2000 field. His areas of expertise include IBM

@server xSeries systems management, Citrix Metaframe and TCP/IP communication.

KyungHa Kim is an Advisory IT Specialist working for IBM Korea since June 1996. Her areas of expertise include providing technical support on AIX platform. Her mission includes various @server pSeries benchmark tests, performance tuning, troubleshooting, and ISV support. She holds a degree in Mathematics.

Matt Robbins is an @server pSeries Technical Sales Specialist in Dallas, Texas. He has over six years of experience working with pSeries systems and AIX. His areas of expertise include UNIX, TCP/IP, and designing e-business solutions for Internet security and Web traffic. He attended the University of North Texas as a student of computer science.

SeungLul Son is an Advisory Education Specialist in IBM Korea. He is an MCSE and CCNA with five years of experience in Microsoft Operating Systems and AIX field. His areas of expertise include Windows NT/2000, UNIX, TCP/IP and Internetworking between different Operating Systems and network devices.

Thanks to the following people for their invaluable contributions to this project:

International Technical Support Organization, Austin Center

Matthew Parente

We would also like to thank the authors of the previous version of this publication:

Laurent Vanel, Leonardo Antonelli, Angela Keelan, Miha Music

We would also like to thank the authors of the original version of this publication:

Yves Bex, John Brantly, Rolf Berger, Dana Lloyd, Alberto Miglioli, Franz Kraemer, Zoran Gagic, Nobuhiko Watanabe

Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Please send us your comments about this or other Redbooks in one of the following ways:

- Fax the evaluation form found in “IBM Redbooks review” on page 549 to the fax number shown on the form.
- Use the online evaluation form found at **ibm.com**/redbooks
- Send your comments in an Internet note to redbook@us.ibm.com

Chapter 1. Operating system concepts

The purpose of this chapter is to introduce some concepts in the operating system design area. There are several ways to structure and organize an operating system. A description of different operating system models will follow shortly.

All these operating systems have one thing in common: They all run tasks in user mode and kernel mode. Tasks that run in kernel mode can access system hardware and system data. Only specific codes are able to run in kernel mode. Applications usually run in user mode.

1.1 Monolithic model

When designing an operating system, one approach is to build a monolithic structure. This approach was used in MS-DOS and early UNIX systems. In this approach, the system consists of functions or procedures that can call any other procedure. The system is not modular enough to easily allow updating of one procedure without updating others. Also, since the bulk of the operating system code runs in the same memory space, it is possible for a component of the operating system to corrupt data being used by other components. Maintenance of such an operating system is also very difficult. Figure 1 on page 2 shows a monolithic operating system.

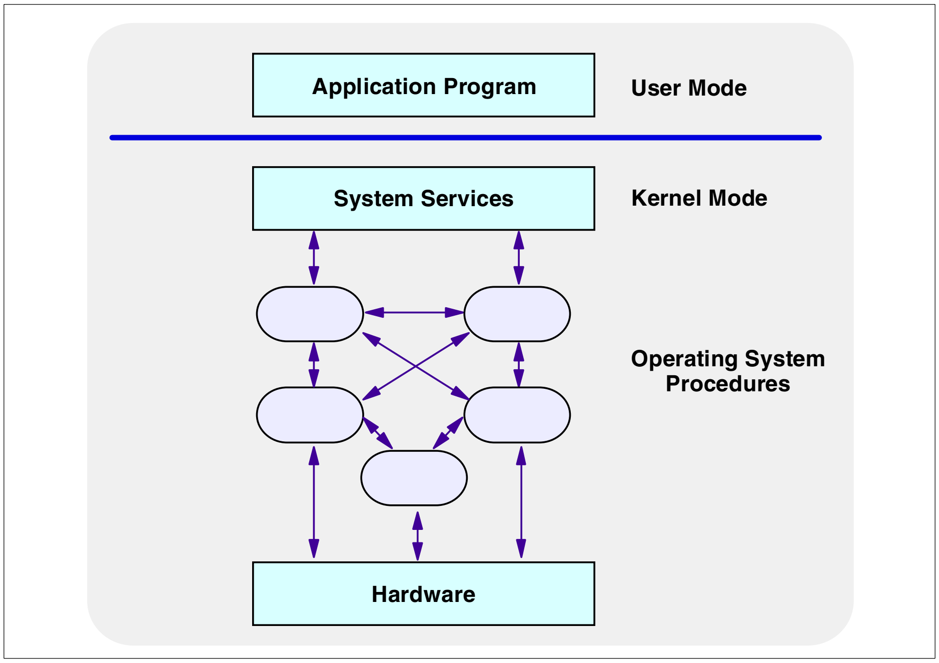


Figure 1. Monolithic operating system

1.2 Layered model

Another way of structuring an operating system is to organize it into layers of code. Code in any particular layer only calls code in lower layers. The higher layer of code only has access to the lower-level interfaces and data structures.

Commands are only passed down to the lower layers, not upward. The advantages of such a hierarchical structure are easy maintenance and debugging.

Maintenance is easier because one entire layer can be replaced without affecting the other layers. Also, each layer can be traced and debugged from the bottom up until the system works correctly. Figure 2 on page 3 shows a picture of a layered operating system:

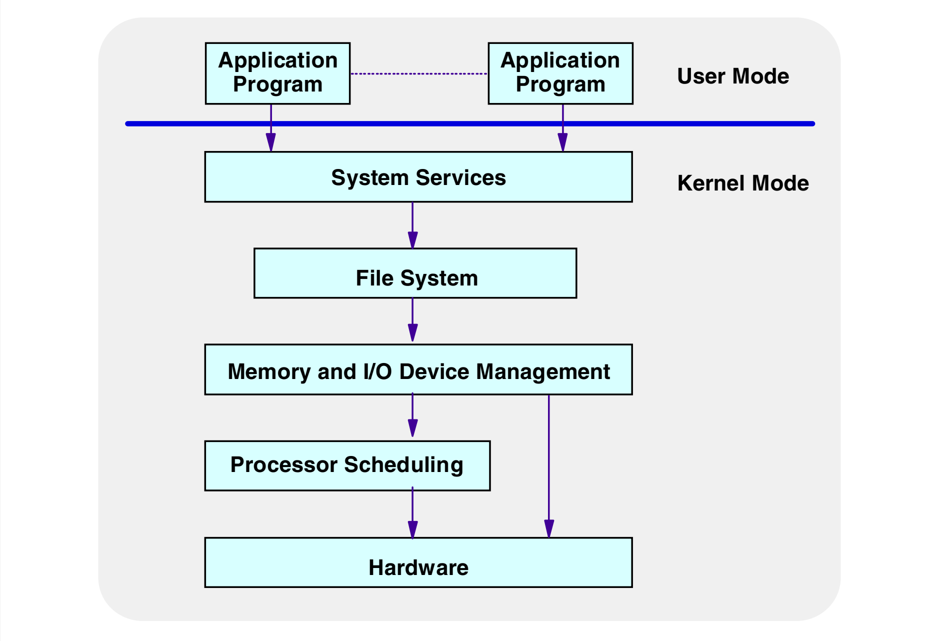


Figure 2. Layered operating system

1.3 Micro kernel

In a traditional operating system, operating system services, such as process management, virtual memory management, network management, file system services, and device management, are built into the kernel. The result is that the operating system is difficult to enhance and difficult to port to a different hardware platform.

A micro kernel provides only the basic operating system services, such as task and thread management, inter process communication, virtual memory services, input/output (I/O) and interrupt services. These services are made available to the user-level tasks through a set of micro kernel interface functions.

Services, such as file systems, network services, and device drivers, operate outside the micro kernel as user-level servers. This makes the kernel smaller. A micro kernel also provides a clean separation of machine-dependent and machine-independent code; it is easily portable to a different hardware platform.

Machine-dependent and device-dependent code is located in the micro kernel.

Also, in such a design, device drivers operate at the user process level above the kernel. This means device drivers can be developed and debugged just like user programs. This also increases portability to another hardware platform.

1.4 Client/server model

In this model, the operating system is divided into several processes. Each process implements a single set of services, such as memory services, process creation services, or processor scheduling services. These server processes run in user mode while waiting for client requests. Clients can be other operating system components or application programs. Clients request services by sending a message to the server. The operating system kernel running in kernel mode delivers the message to the server that performs the operation. Then, the kernel sends the results back to the client in another message.

In this approach, the operating system components can be small. Also, since each server runs in a separate user-mode process, a single server can fail without crashing the rest of the operating system. Figure 3 on page 5 depicts the client/server operating system model:

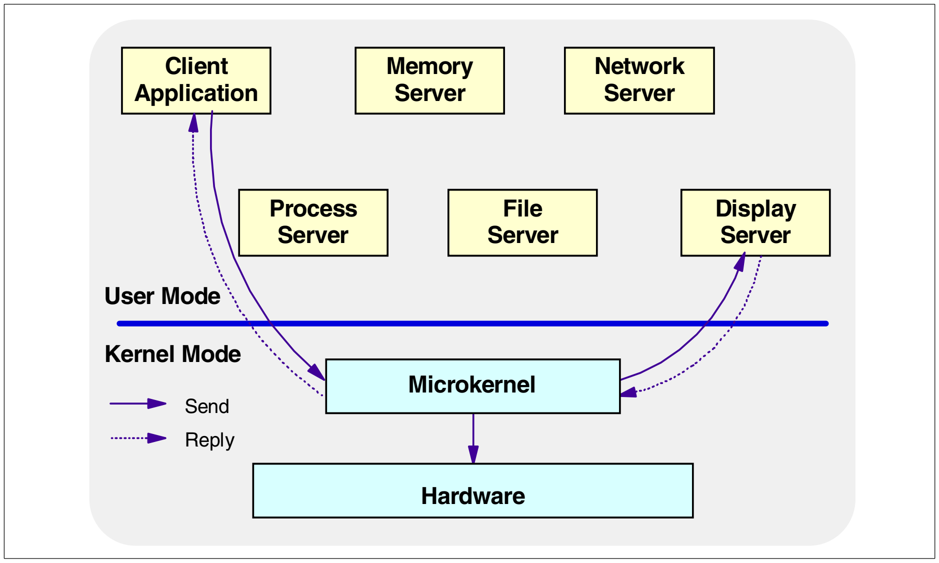


Figure 3. Client/server operating system

Chapter 2. Operating system architecture

The purpose of this chapter is to give an overview of AIX 5L and Windows 2000 architecture and design. The history section of this introduction will help you understand the design choices of these operating systems.

2.1 AIX architecture

It is well beyond the scope of this book to completely explain the entire AIX architecture and kernel in great detail. Instead, the topic will be discussed at the component level.

2.1.1 AIX Version history

AIX stands for Advanced Interactive eXecutive and it is IBM's flavor of UNIX. AIX was created as IBM's premier UNIX operating system for their line of RISC technology (RT) servers in the mid 1980's. Originally, AIX was primarily based on AT&T's UNIX System Version 2, but as it has evolved over the years through different versions, it has taken on characteristics of the UNIX Berkeley Software Distribution (BSD UNIX), the OSF/1 version, and versions of UNIX that have come from the Open Software Foundation (OSF, now Open Group), of which IBM was a founding member.

2.1.1.1 AIX Version 3

First released in February of 1990, AIX Version 3.0 through 3.2.5 was created to support IBM's Reduced Instruction Set Computing (RISC) line of Power servers. It was the first version of AIX to offer POSIX IEEE 1003.1-1988 standards conformance, X/Open XPG3 base level compliance, and Berkeley Software Distribution 4.3 (4.3 BSD) compatibility.

Also, the operating system as a whole took on new tools and enhancements not offered by other flavors of UNIX.

Logical Volume Manager (LVM)

The Logical Volume Manager (LVM) introduced a hierarchical storage management system to AIX. LVM introduced the concept of "logical volumes" to AIX storage management, and allowed a more dynamic configuration of physical partitions that allowed system data to span several physical disks.

System Management Interface Tool (SMIT)

To provide an easier and more user friendly interface to AIX SMIT was created as a menu driven tool to execute support for installation, configuration, device management, problem determination, and storage management. Through a series of interactive menus and dialogs, SMIT automatically builds, executes, and logs the appropriate AIX system commands required to execute the required operation.

Trusted Computing Base (TCB)

The TCB within AIX offers a means to restrict access of system resources in a secure manner to authorized users and processes. TCB also allows for system auditing and event logging of suspicious system events, and allows for a system administrator to make sure that system resources are only being used along his or her security parameters.

Transmission Control Protocol/Internet Protocol (TCP/IP)

TCP/IP support is an integral part of all versions of AIX and provides network connectivity and application level interoperability with other computer systems over local area, wide area, and asynchronous networks.

Motif X Window manager

Most flavors of UNIX provide for some sort of graphical user interface (GUI), and in early versions of AIX, this was done through the Motif X Window manager. Motif provided a fully configurable and programable graphical user interface to AIX and support for Motif became integrated into the AIX Window functionality.

Network File System (NFS)

Although it was originally only offered as a separately licensed program before AIX 3.2, the Network File System eventually became an integral part of AIX and was integrated into the AIX 3.2 offering. NFS allows for local mounting of non-local storage media over a TCP/IP network.

2.1.1.2 AIX Version 4

In July of 1994, IBM introduced AIX Version 4. Throughout AIX Version 4, AIX saw many changes and enhancements to the system kernel and in the following sections, we shall cover the changes AIX went through from AIX 4.1 through AIX Version 4.3.3.

Network Install Manager (NIM)

Network installs were possible within previous versions of AIX, but it became a formal and fully supported process with AIX Version 4 through the Network Install Manager. NIM installs the basic operating system and other operating system components from the server onto clients within the network. NIM streamlined the install process for AIX most especially on the SP hardware platform where many AIX installs may have to take place and allows these installs to take place without constant system administrator intervention.

Journaled File System (JFS)

Before AIX Version 4.1, data was written within logical volumes to the file system in set blocks of 4096 bytes. With the introduction of a journaled file system into AIX, support for data block fragments as small as 512 bytes was created. This allows for files to more efficiently utilize disk space when a data file is smaller than 4096 bytes long.

Dynamic Host Configuration Protocol (DHCP)

Support was added for DHCP in AIX Version 4.1.4. DHCP is a network service under TCP/IP that allows for automatic network configuration of network clients upon start up. The system administrator then only has to configure one server on the network with all the relevant network data. Clients can then access a host configuration automatically from this server upon bootup. For information on DHCP support on AIX, please read *Beyond DHCP - Work Your TCP/IP Internetwork with Dynamic IP*, SG24-5280-01.

Common Desktop Environment (CDE)

The AIX Common Desktop Environment replaced the Motif X Window manager as an industry standard graphical user interface to AIX. CDE 1.0 became the default bootup desktop in AIX Version 4.1.3 and was included in both the AIX Version 4 for Clients package and AIX Version for Servers package.

Support for Symmetric Multi-Processing (SMP) systems

AIX Version 4 was the first version of AIX to support systems with multiple processors. Also, changes and additions were made to the system kernel and system components to optimize AIX for SMP architecture systems and multi-threaded applications. Support for SMP hardware and introduction of threads into AIX kernel.

Web-based System Manager

Web-based System Manager enables a system administrator to manage an AIX machine either locally from a graphics terminal or remotely from a PC or IBM @server pSeries client. Information is entered through the use of GUI components on the client side. The information is then sent over the network to the Web-Based System Manager server, which runs the commands necessary to perform the required action.

Workload Manager (WLM)

The Workload Manager (WLM) is designed to give the system administrator greater control over how the scheduler and Virtual Memory Manager (VMM) allocate CPU and physical memory resources to processes. This can be used to prevent different classes of jobs from interfering with each other and to allocate resources based on the requirements of different groups of users.

The major use of WLM is expected to be for large SMP systems, typically used for server consolidation, where workloads from many different server systems, such as print, database, general user, transaction processing systems, and so on, are combined. These workloads often compete for resources and have differing goals and service level agreements. At the same time, WLM can be used in uniprocessor workstations to improve responsiveness of interactive work by reserving physical memory. WLM can also be used to manage individual SP nodes.

Another use of WLM is to provide a buffer between user communities with very different system behaviors. WLM can help prevent effective starvation of workloads with certain behaviors, such as interactive or low CPU usage jobs, from workloads with other behaviors, such as batch or high CPU usage.

WLM gives the system administrator the ability to create different classes of service, and specify attributes for those classes. The system administrator has the ability to classify jobs automatically to classes based upon the user, group, or path name of the application.

Internet Protocol version 6 support (IPv6)

Internet Protocol Version 6 (IPv6) is the next generation Internet Engineering Task Force (IETF) networking protocol that will become the industry standard network protocol for the Internet of the future.

IPv6 extends the maximum number of Internet addresses to handle the ever-increasing Internet user population. IPv6 is an evolutionary change from IPv4 and has the advantage of allowing a mixture of the new and the old to coexist on the same network. This coexistence enables an orderly migration from IPv4 (32-bit addressing) to IPv6 (128-bit addressing) on an operational network.

Online documentation libraries

AIX Version 4.3.3 saw the introduction of the extended Documentation Library Service to integrate the navigation, reading, and search of online documents. You can use these functions with a new documentation library Graphical User Interface (GUI). This new GUI offers easier access to online documentation with a single integrated graphical user interface.

The AIX operating system documentation can be accessed through this library service. Additionally, you can register locally written HTML documents into the library so that you can go to a single library GUI to access a wide range of documents. You can have a unified presentation of documents to users so they will only need to use one library application to find any HTML documentation that is stored on the system.

For example, in addition to AIX documentation, other documents could include online documentation for customer applications and also company policies and procedures. The library services can be made available locally, or through use of a Web server, and the documents can be used remotely by an intranet capable client (AIX or PC).

64-bit computing

64-bit computing was introduced with AIX 4.3.0 for support of the new RS64 processors, and this was the first version of AIX to include 64-bit processing support. Furthermore, the kernel kept binary compatibility with previous versions of AIX and retained support for 32-bit applications and was even capable of running 32-bit and 64-bit applications simultaneously. This allowed those using AIX to take advantage of 64-bit computing without having to abandon their existing 32-bit solutions.

The advantages with 64-bit computing include the ability to perform extended precision arithmetic, being able to handle numbers up to 64 bits long in one computing cycle, and greater ability to address data in memory and in storage.

2.1.1.3 AIX 5L

AIX 5L was created through Project Monterey, a coalition of IBM, SCO, and Intel. You can read about Project Monterey at:

`http://www.projectmonterey.com/`

This coalition was created with the intention of creating an open standards version of UNIX compatible with 64-bit based hardware platforms. Along with being 64-bit compliant, AIX 5L also offers affinity with another open standards version of UNIX, Linux, and has binary compatibility with previous versions of AIX. AIX 5L Version 5.0 has many enhancements and additions over the previous version, AIX 4.3.3. For full documentation on the additions and enhancements made to AIX 5L, please read *AIX 5L Differences Guide,* SG24-5765-00.

Journaled File System (JFS2)

The Journaled File System 2 (JFS2) is an enhanced and updated version of the JFS on AIX Version 4.3 and previous releases. Journaled File System 2 (JFS2) is intended to provide a robust, quickly restartable, transaction-oriented, log-based, and scalable byte-level file system implementation for AIX environments. JFS2 has new features that include extent based allocation, sorted directories, and dynamic space allocation for file system objects. While tailored primarily for the high throughput and reliability requirements of servers, JFS2 is also applicable to client configurations where performance and reliability are desired.

Both JFS (the default) and JFS2 are available on POWER systems. Only JFS2 is supported on Itanium-based systems.

NFS statd multithreading

In AIX 5L, the NFS statd daemon is multi-threaded. In AIX Version 4.3, when the statd daemon is detecting whether the clients are up or not, it hangs and waits for a time out when a client can not be found. If there are a large number of clients that are offline, it can take a long time to time out all of them sequentially.

With a multithreading design, stat requests run in parallel to solve the time-out problem. The server statd monitors clients and the client's statd monitors the server if a client has multiple mounts. Connections are dropped if the remote partner cannot be detected without affecting other stat operations.

Passive write mirror consistency

AIX 5L introduces a new passive mirror write consistency check (MWCC) algorithm for mirrored logical volumes. This option only applies to big volume groups.

Previous versions of AIX used a single MWCC algorithm, which is now called the active MWCC algorithm to distinguish it from the new algorithm. With active MWCC, records of the last 62 distinct logical transfer groups (LTG) written to disk are kept in memory and also written to a separate checkpoint area on disk. Because only new writes are tracked, if new MWCC tracking tables have to be written out to the disk checkpoint area, the disk performance can degrade if there are a lot of random write requests issued. The purpose of the MWCC is to guarantee the consistency of the mirrored logical volumes in case of a crash. After a system crash, the logical volume manager will use the LTG tables in the MWCC copies on disk to make sure that all mirror copies are consistent.

The new passive MWCC algorithm does not use an LTG tracking table, but sets a dirty bit for the mirrored logical volume as soon as the volume is opened for writes. This bit gets cleared only if the volume is successfully synced and is closed. In the case of a system crash, the entire mirrored logical volume will undergo a background re-synchronization that is spawned during the vary-on of the volume group, because the dirty bit has not been cleared. Once the background re-synchronization completes, the dirty bit is cleared, but can be reset at any time if the mirrored logical volume is opened.

It should be noted that the mirrored logical volume can be used immediately after system reboot, even though it is undergoing background re-synchronization.

The benefit of the new passive MWCC algorithm (as compared to the default active MWCC algorithm) is better performance during normal system operations. However, there is an additional I/O that may slow system performance during the automatic background re-synchronization that occurs during recovery after a crash.

Configuration manager

The installation of new hardware has been streamlined in AIX 5L through enhancements to the configuration manager (cfgmgr). It now adds new devices in parallel, whereas previous versions of cfgmgr added devices sequentially as it discovered them.

Web-based System Manager

The Web based System Manager tool has the following enhancements over AIX Version 4.3.3:

- A new management console
- Point-to-point multiple host management
- New Java 1.3 compliance
- Shell script and API execution interface
- Dynamic user interface
- Kerberos v5 integration
- Integration with the new Resource monitoring and control subsystem, introduced in section 2.1.1.4, “New in AIX 5L” on page 14

Workload Manager (WLM)

Workload Manager (WLM) also has some new enhancements in AIX 5L, these include:

- Graphical display of resource utilization
- Integration of Performance Toolbox with WLM classes
- Management of disk I/O bandwidth
- Dynamic access to configuration files, allowing the system administrator to change WLM configuration on the fly

2.1.1.4 New in AIX 5L

On all UNIX systems, there must be at least one file system, called the root file system or /, within which the other file systems can be accessed on the local system. In AIX 5L, several other file systems are created at installation time: /usr, /var, /tmp, /home, /opt, and /proc.

The /proc file system

New in AIX 5L, the /proc file system contains a directory for each kernel data structure and active process running on the system.

Each of these entries gets a Process Identification Number (PID) within the kernel memory, and now within AIX 5L each PID gets its own directory structure within /proc. Working with kernel data structures and processes in this manner allows a debugger or system administrator to stop and start threads within a process, trace syscalls, trace signals, and read and write to virtual memory within a process. The new /proc file system can be invaluable in debugging system processes and applications.

The /opt file system

Also new to AIX in 5L is the /opt or “optional” directory. This directory is reserved for the installation of add-on application software packages and is integral to AIX 5L’s new affinity with Linux applications.

Deactivating Active Paging Space

This command provides new flexibility (does not require rebooting) when changing configurations, moving paging space to another device, or dividing paging space up between drives. Until this release, allocated and activated paging space must stay active until the next re-boot. With this release, paging space can be deactivated without rebooting by using the new `swapoff` command. The new `shrinkps` command creates a new, temporary space, deactivates the original, changes the original to be smaller, reactivates it, and then deactivates the temporary space and returns it to logical volume status.

Resource monitoring and control (RMC)

Resource monitoring and control (RMC), comparable to Reliable Scalable Cluster Technology (RSCT) on the SP, allows a system administrator to configure an AIX 5L system to monitor itself in terms of performance and availability and respond. The RMC subsystem comes pre-configured with 84 conditions and eight responses that can be used as is or as templates for creating your own performance monitoring conditions and responses.

Native Kerberos 5 support

The AIX 5L operating system allows the system administrator to replace the default login process with Kerberos 5 authentication. Kerberos 5, once a user has logged in their ID, will acquire all appropriate network and system credentials. In previous AIX releases, the distributed computer environment (DCE) and the network information system (NIS) were supported as alternate authentication mechanisms. AIX Version 4.3.3 added Lightweight Directory Access Protocol (LDAP) support and the initial support for specifying a loadable module as an argument for the user/group managing commands, such as `mkuser`, `lsuser, rmuser`. But this was only documented in the /usr/lpp/bos/README file. AIX 5L is now offering a general mechanism to separate the identification and authentication of users and groups, and defines an application programming interface (API) that specifies which function entry points a module has to make available in order to work as an identification or authentication method. This allows for more sophisticated customized login methods beyond what is provided by the ones based on /etc/passwd or DCE.

Virtual IP address support

For applications to get access to communication and network services, previous releases of AIX required applications to bind themselves to a literal network interface. With the application bound to a literal IP address, the application could become inaccessible if the IP interface went down or TCP/IP services became interrupted.

With the addition of virtual IP address (VIPA) support in AIX 5L, an application can be bound to a virtual IP address that can be routed to any accessible hardware network interface. This way, if one interface goes down, the VIPA can be routed to another interface, and if done fast enough, can prevent the loss of TCP/IP sessions. Furthermore, a VIPA can be brought down independently of the access of other running applications. This allows multiple applications to use the same interface for communication and for the virtual IPs to be brought up or down without affecting any other application's network interface.

2.1.2 Architecture description

The AIX operating system is a layered operating system. In a layered operating system, particular system functions are thought of as belonging to a specific layer or level of the total system. Each layer can only communicate with an adjoining layer through a predefined mechanism. This layering simplifies the design process and allows for simplified debugging because you only concentrate your focus on one layer at a time. In addition, a specific layer of the operating system can be added, upgraded, or completely replaced with another layer without having to rewrite existing layers.

The addition of the Platform Abstraction Layer (PAL), added in AIX Version 4.2, is a good example of the ability to add layers to an existing operating system.

With the release of AIX Version 4.3 came 64-bit computing. With a number of models in the IBM @server pSeries range using 64-bit processors, AIX Version 4.3 is fully capable of running both 32-bit and 64-bit applications seamlessly.

2.1.2.1 64-bit computing

It is important to note that the 64-bit execution environment for application processes is an upward-compatible addition to AIX capability, not a replacement for the existing 32-bit function. The design IBM chose for 64-bit AIX allows existing 32-bit applications to run with 64-bit applications without any changes, thus protecting the investment users have made in their current applications. Users can take advantage of the features of 64-bit AIX when business needs dictate.

While there is no formal definition of 64-bit computing, it is distinguished from 32-bit computing by the following:

- **Large file support** - This ability to address data in fields larger than 2 GB requires a program be able to specify file offsets larger than a 32-bit number. This capability is generally considered to be a 64-bit function, even though it does not require 64-bit hardware support. AIX Version 4.2 provided this capability for 32-bit programs, and AIX Version 4.3 provides it for 64-bit programs as well. Since it does not depend on 64-bit hardware, this function can be used on any IBM @server pSeries system running the appropriate release of AIX.
- **Very large memory support** - The new dimension of scalability introduced by 64-bit technology is the opportunity for some programs to keep very large amounts of data in memory, both resident in physical memory and accessible in their 64-bit virtual memory address space.
- **Large application virtual address space** - In 32-bit systems, an individual program, or process, may have between 2 GB and 4 GB of virtual address space for its own use to contain instructions and data. With 64-bit computing, applications may run in a 64-bit address space, where an individual program’s addressability becomes measured in terabytes (TB).
- **64-bit integer computation, using hardware with 64-bit general purpose registers** - Native 64-bit integer computation is provided by 64-bit hardware and is utilized by programs computing on 64-bit data types. While there are some applications that need to do computations on

integer numbers larger than 2^{32}, the key benefits of this capability are in performing arithmetic operations on pointers in 64-bit programs. Floating point computation already includes 64-bit precision on all IBM @server pSeries systems.

At this stage, it is not a requirement of 64-bit computing that 64-bit hardware be utilized.

For more information on the IBM @server pSeries 64-bit technology, see the white paper entitled *The RS/6000 64-bit Solution* online at:

`http://www.rs6000.ibm.com/resource/technology/64bit6.html`

2.1.2.2 The AIX kernel

At the very heart of the AIX operating system is the AIX kernel. The AIX kernel provides the ability to share system resources simultaneously among many processes or threads and users. The most important resources that the kernel manages are the processor(s) (CPUs), memory, and devices. By careful design, the kernel is preemptable and pageable, yet it is also dynamic and extendable.

Preemptable

The kernel can be in the middle of a system call and be preempted. This preemption could signal a context switch that causes an entirely new thread of execution inside the kernel. Threads are assigned a priority by the kernel that the kernel can adjust based on certain factors, such as the length of time the thread has been running. Preemptability allows the kernel to respond to real-time processes much faster than other operating systems.

In a preemptable kernel, a higher-priority thread that becomes runable may preempt a low-priority thread even though it is executing kernel code. Device drivers and other interrupts can preempt processes in kernel mode, but, upon its return from the interrupt, the preempted process retains control of the CPU. In contrast, processes in user mode are always preemptable. Upon its return from an interrupt, the kernel decides which process should run next, based on priority.

Pageable (demand paging)

A pageable kernel means that only those parts of the kernel that are being used or referenced are kept in physical memory. Kernel pages that have not been used recently can be paged out. Some parts of the kernel do not get paged out. Instead, they are pinned. An example of pinned kernel code is the interrupt processing section of the device drivers.

The kernel utilizes a pager daemon to keep a pool of physical pages free. It uses a Least Recently Used (LRU) algorithm. If the number of pages available goes below a high-watermark threshold, the pager frees the oldest LRU pages until a low-watermark threshold is reached.

In other operating systems, including some UNIX variants, the entire kernel must be loaded and pinned into memory. This feature acquires more significance when you consider that AIX functionality can be dynamically extended using kernel extensions. Therefore, while the AIX kernel may tend to be larger than other kernels, due in part to user-added functions through kernel extensions, it usually requires less physical memory to actually run.

Dynamic

In AIX, kernel extensions can be added to and deleted from the kernel as needed. This allows an administrator to add new device drivers, file systems, and other kernel code at any time without having to recompile the kernel. Since recompiling the kernel is not required, rebooting the system is not normally required in order to make changes take effect.

Extendable

Kernel extensions are dynamically loadable in AIX. These extensions allow programs direct access to kernel resources for better performance. A system programmer can add new services in AIX by making use of the defined kernel extension types. These extension types, often referred to as hooks, can be divided into four categories:

- Device drivers
- System calls
- Virtual file systems
- Kernel extension and device driver management routines

When properly coded, kernel extensions add extensibility, configurability, and ease of system administration to AIX.

This combination of features allows the AIX kernel to be highly scalable, from the smallest of the PowerPC processors to the largest Scalable POWERparallel (SP) systems. Yet, it also allows for the fine tuning of an operating environment.

64-bit Kernel

AIX 5L provides a scalable, 64-bit kernel capable of supporting increased system resources and much larger application workloads on 64-bit hardware. In addition, the 64-bit kernel offers scalable kernel extension interfaces, allowing kernel extensions and device drivers to make full use of the kernel’s system resources and capabilities. The expanded capabilities of the 64-bit AIX 5L kernel improve the ability to run an expanding application workload on a single system. Kernel extensions and device drivers must be compiled in 64-bit mode to be loaded into the 64-bit kernel. The 64-bit AIX 5L kernel provides the environment for porting and developing kernel extensions. The 64-bit AIX 5L kernel is the only kernel provided on Itanium-based systems. For POWER, both 32-bit and 63-bit kernels are provides. The 64-bit kernel can be installed and enabled on most 64-bit machines.

Figure 4 shows the combinations of application, kernel, and hardware related with 32-bit and 64-bit kernel mode.

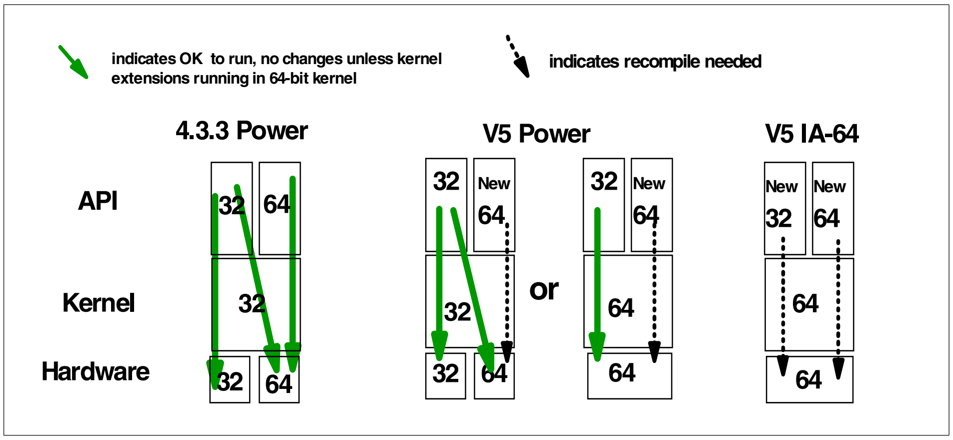

indicates OK to run, no changes unless kernel extensions running in 64-bit kernel
indicates recompile needed
4.3.3 Power
V5 Power
V5 IA-64
API
Kernel
Hardware
32
64
32
32
64
32
New
64
32
32
64
or
32
New
64
64
64
New
32
New
64
64
64

Figure 4. Application, Kernel, and Hardware combinations

Power 32-bit V4 applications run on AIX 5L POWER “as is” and a one time recompile is required for 64-bit applications on AIX 5L POWER.

64-bit Application Scalability

AIX 5L provides a more scalable application binary interface (ABI) for 64-bit applications. This scalability is provided by changing the sizes of some fundamental data types for 64-bit applications and will allow these applications to take advantage of expanded capabilities of the AIX 64-bit kernel. The scalable 64-bit ABI will be supported by the 32-bit kernel as well as the 64-bit kernel.

2.1.2.3 Modes of operation (execution modes)

There are basically two modes of operation in the AIX operating system: the kernel mode and the user mode. Kernel extensions run in the kernel mode and user applications tend to run in the user mode. Kernel mode has unlimited access to these and other functions, including additional instructions/commands. In user mode, programs have limited access to kernel data and global structures .

The receipt of an interrupt can cause a process or thread running in user mode to change to the supervisor state in order to handle the exception. An interrupt can be caused by an external signal, program error, program request, or any other unusual condition. The receipt of an interrupt causes a switch to the supervisor state and an immediate branch to a specific memory location or vector. The operating system has code at that vector to save the machine state and branch to a handler routine.

Note

Often, the terms *exception* and *interrupt* get confused. Generally, the correct term in AIX terminology is interrupt.

2.1.2.4 AIX 5L kernel subsystems

Figure 5 on page 21 describes the AIX kernel architecture.

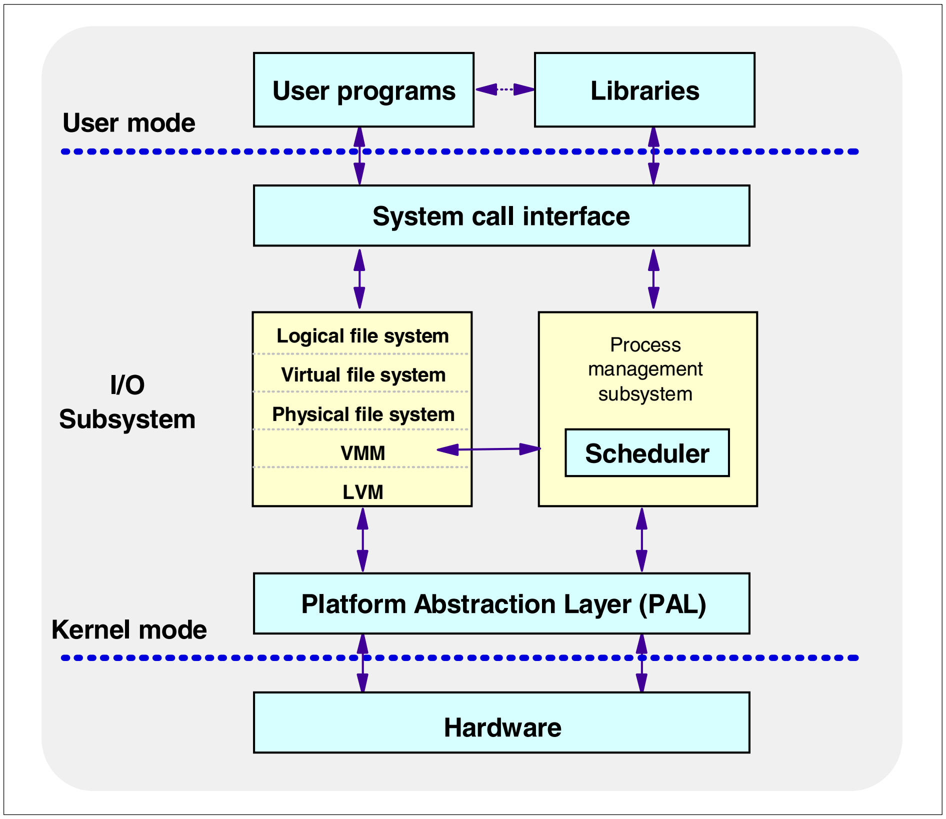


Figure 5. AIX 5L kernel architecture

Within the kernel, there are various subsystems that are dedicated to particular functions. These subsystems generally operate with a very high priority in the operating system, as do most kernel processes. The following sections list the major components of the kernel:

System call interface

The system call interface is the primary mechanism for user code and applications to access the kernel. This layer can be thought of as the Application Program Interface (API) to the kernel. Applications make system calls to get information, perform operations, and access resources through the kernel.

I/O subsystem

Access to files and directories is controlled by various layers within the I/O layer of the AIX kernel. There are many parts of this I/O layer and would take volumes to describe. However, the major functions contained in the I/O layer are meant to provide a consistent view to the user of any file within the operating system, whether it is a real physical file, a remote file, or even a logical file. The intent is to be able to deal with all file types via the same system calls, such as open(), close(), read (), write(), and so on.

The following is a listing of the major functions of this layer with a brief description of each:

- **Logical File System (LFS)** - The LFS provides AIX and user applications with a consistent view of all file system implementations. Physical file system types are shielded by the logical file system.
- **Virtual File System (VFS)** - The VFS provides a standard set of operations on entire file systems.
- **Physical File System (PFS)** - There are different types of PFSs, such as Journaled File System, CD-ROM file system, NFS, and so on.
- **Virtual Memory Manager (VMM)** - The VMM provides the processes with a virtual address space allowing the creation of memory segments that are greater than the physical memory.
- **Logical Volume Manager (LVM)** - The LVM provides the definition and management of volume groups, logical volumes and physical volumes. This creates a virtual disk environment that can be dynamically changed.

Process management (scheduler)

A thread is the smallest schedulable entity within AIX 5L. A process, made with a single thread or a collection of threads, is a self-contained entity that consists of the information required to run a program. Process management allows many processes, or threads, to exist simultaneously and share the CPU. The scheduler decides which thread runs next and, in a multiprocessor machine, decides on which processor the thread will run.

The AIX scheduler uses a time-sharing priority-based scheduling algorithm. The scheduler periodically scans the list of all active processes/threads and recalculates process priorities based on the initial priority and the amount of processor time used. It tends to favor processes that do not consume large amounts of the processor time because the amount of processor time used by the scheduler is included in the priority recalculation equation.

Virtual memory manager (VMM)

Virtual memory is a mechanism by which the real memory available for use appears larger than its true size. The virtual memory system is composed of physical disk space where portions of a file that are not currently in use are stored, as well as the system’s real memory. The physical disk part of virtual memory is divided into three types of segments that reflect where the data is being stored:

- Local persistent segments from a local file system
- Working segments in the paging space
- Client persistent segments from CD-ROM or remote file systems

One of the basic building blocks of the AIX memory system is the segment, which is a 256 MB (2^{28}) piece of the virtual address space. Each segment is further divided into 4096 byte pages of information. Each page sits in a 4 KB partition of the disk, known as a slot. Similarly, real memory is divided into 4096 byte page frames. A frame (or page frame) usually means a physical memory page as opposed to a virtual page; the context of its use usually indicates which one is intended. When a page is needed from its disk location, it is loaded into a frame in real memory.

The Virtual Memory Manager (VMM) coordinates and manages all the activities associated with the virtual memory system. The VMM is responsible for allocating real memory page frames and resolving references to pages that are not currently in real memory.

Previous releases of AIX managed all of a system’s real memory as one large resource that was available for the programs executing in one or more CPUs to address and use through the VMM. Thus, there was one list of free memory frames, and a one page replacement daemon that would help ensure that the required pages actually could be located in system RAM.

Since systems continue to grow, AIX has improved memory management through the use of multiple free frame lists and multiple page replacement daemons. This increases the VMM concurrency since contention has been reduced in the serialization mechanisms and processes with lower latencies can now service the memory requests.

A memory pool is a range of memory on which a single memory replacement operation operates; that is, only one LRU (Least Recently Used) manages this pool of memory frames. A memory frame (or page frame) appears in one, and only one, memory pool. A frame set is a set of memory frames; the frames in a set belong exclusively to that set.

Using this terminology, previous releases of AIX can be considered to consist of one memory pool and one frame set. Now, AIX 4.3.3 and the higher can have all of its system-wide memory frames managed with multiple memory pools. Each pool has roughly the same number of frames (so that the system is balanced) and the frames in a pool are organized in multiple frame sets.

The number of frame sets and the number of memory pools depend on the number of CPUs and the amount of real memory in the system. In the current implementation, an LRU daemon called lrud (least recently used daemon) is created and started at the end of the VMM initialization for each memory pool.

As mentioned earlier, the number of memory pools depends on the number of CPUs and the amount of real memory on your system. For an MP kernel (packaged in the bos.mp fileset), there should be at least one lrud daemon, even if it is running on a single CPU system. With the UP kernel (in the bos.up fileset), there is only one memory pool and one frameset. There will never be an lrud when the UP kernel is active.

Now, if a thread needs some page frames, it is no longer constrained by having only one lrud available to check memory in an MP environment. Relevant VMM locks have also been enhanced.

Platform Abstraction Layer (PAL)

The PAL is a layer in the kernel that has been introduced with AIX Version 4. It is, ostensibly, the hardware-control layer. This is the layer in which device drivers are accessed by the kernel. All hardware-dependent code has been extracted from the kernel and placed in kernel extensions. This facilitates new hardware and device support.

2.1.2.5 Multitasking and multithreading support

AIX 5L is a multitasking system that uses processes and threads. A thread is an independent flow of control that operates within the same address space as other independent flows of control within a process. In previous versions of AIX, and in most UNIX systems, thread and process characteristics are grouped into a single entity called a process. In other operating systems, threads are sometimes called *lightweight processes* or the meaning of the word *thread* is sometimes slightly different.

In traditional single-threaded process systems, a process has a set of properties. In multi-threaded systems, these properties are divided between processes and threads.

A process in a multi-threaded system is the changeable entity and must be considered as an execution frame. It has all traditional process attributes, such as process ID, process group ID, user ID, group ID, environment, and working directory.

A process also provides a common address space and common system resources for:

- File descriptors

- Signal actions
- Shared libraries
- Inter-process communication tools, such as message queues, pipes, semaphores, or shared memory

A thread is the schedulable entity. It has only those properties that are required to ensure its independent flow of control. A kernel thread is a kernel entity, such as processes and interrupt handlers; it is the entity handled by the system scheduler. A kernel thread runs within a process but can be referenced by any other thread in the system. The programmer has no direct control over these threads unless he or she is writing kernel extensions or device drivers.

A user thread is an entity used by programmers to handle multiple flows of controls within a program. The API for handling user threads is provided by a library called the threads library. A user thread only exists within a process; a user thread in process A cannot reference a user thread in process B. The library uses a proprietary interface to handle kernel threads for executing user threads. The user threads' API, unlike the kernel threads' interface, is part of a portable programming model. Thus, a multi-threaded program developed on an AIX system can easily be ported to other systems.

User threads are mapped to kernel threads by the threads library. The way this mapping is done is called the thread model. There are three possible thread models corresponding to three different ways of mapping user threads to kernel threads:

- M:1 model
- 1:1 model
- M:N model.

The mapping of user threads to kernel threads is done using virtual processors. A virtual processor (VP) is a library entity that is usually implicit. The virtual processor looks like a real processor to the user thread; the vp behaves just as a CPU does for a kernel thread. In the library, the virtual processor is a kernel thread or a structure bound to a kernel thread.

In the M:1 model, all user threads are mapped to one kernel thread; all user threads run on one VP. The mapping is handled by a library scheduler. All user threads programming facilities are completely handled by the library. This model can be used on any system, especially on traditional single-threaded systems. The following Figure 6 on page 26 illustrates this model:

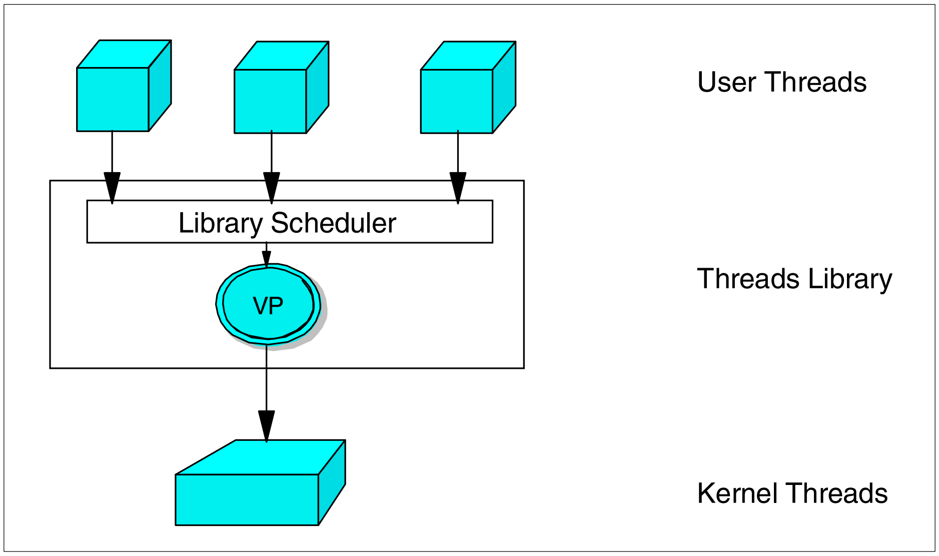


Figure 6. M:1 threads model in AIX 5L

In the 1:1 model, each user thread is mapped to one kernel thread; each user thread runs on one VP. Most of the user threads' programming facilities are handled directly by the kernel threads. Each thread can be separately and independently passed out to any processor on the system for execution. Figure 7 on page 27 illustrates this model.

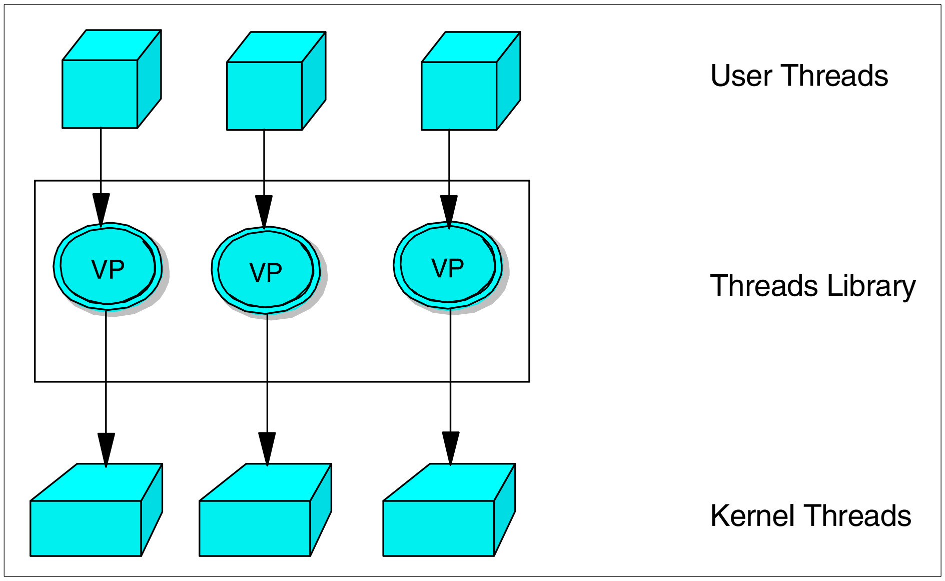


Figure 7. 1:1 threads model in AIX 5L

In the M:N model, all user threads are mapped to a pool of kernel threads and run on a pool of virtual processors. A user thread may be bound to a specific VP, as in the 1:1 model. All unbound user threads share the remaining VPs. This is the most efficient and complex thread model; the user threads' programming facilities are shared between the threads' library and the kernel threads. Figure 8 on page 28 illustrates this model:

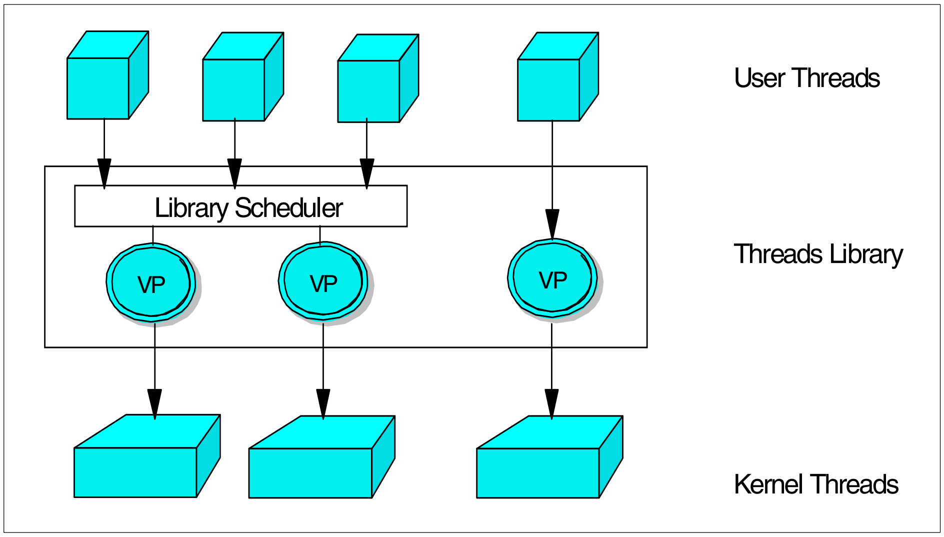


Figure 8. M:N threads model in AIX 5L

By implementing a multi-threaded kernel, AIX is well suited to run in a Symmetric Multiprocessor (SMP) system. AIX has been optimized to run on SMP systems, and scalability on these systems is very high.

2.1.2.6 Multiuser support

AIX has been a true multiuser system from the very beginning; that is, multiple concurrent users are supported.

A number of users can be logged into an AIX system at the same time from various types of devices, such as modems, ASCII terminals, and X-stations.

There is a master user, called root. The root user is basically the administrator of the system. This user can perform system-wide operations, such as installing or removing operating system software and configuring the operating environment. Generally, root is exempt from most system security checking. Therefore, only an experienced and trusted user should be granted access to the root user ID.

As an analogy, any user of a DOS-based system would be similar to the root user of a UNIX system in that he or she would be able to modify or delete any file on the system. Most other users on the system are considered regular users and are usually not able to modify system-related files.

Access to files and directories is controlled via file permissions. Files and directories are owned by a user. A user is also a member of a group or several groups. Groups are a collection of users that can be granted permissions easily. The owner of a directory or file can specify who may read, execute, and write to them. This is standard UNIX behavior.

2.2 Windows version history

The road towards Windows 2000 has been long and winding with both minor and major changes. The following section will discuss the evolvement of Microsoft's operating systems for the Intel platform.

2.2.1 Pre-Windows 2000 operating systems from Microsoft

While operating systems do provide an interface for applications and users, the key purpose of any OS is to enable hardware. In the early 1980s, Microsoft entered the PC operating system arena with its MS-DOS product to enable 8086 architecture hardware. As hardware advanced, Microsoft enhanced MS-DOS to enable the new technologies like larger hard drives, support for more RAM, and higher resolution graphic adapters. The latter spurred Microsoft to develop a graphical shell for DOS called Windows. Windows was not a new OS but merely a graphical user interface (GUI) for MS-DOS.

As hardware advanced over the next few years, it became apparent that a new family of operating systems would be required to enable the powerful machines of tomorrow. It was also apparent that there must be a completely new design strategy compared to the ones of DOS and Windows, which only enabled the Intel family of processors. The result of this new way of thinking was Windows New Technology (Windows NT).

Still, with a big installed base of MS-DOS and Windows, introducing 32-bit extensions to Windows 3.11 in the form of Win32s evolved into a big face-lift for consumer Windows. In 1996, Microsoft introduced Windows 95 and, later, Windows 98; recently Windows Me was released.

Over the course of the MS-DOS evolution, it is an understatement to say that Microsoft has gathered a very large user base on these machines.

Windows NT was created by a carefully selected group of OS developers headed by David Cutler (formerly a DEC employee and a developer of VAX/VMS). Once the group developed a design strategy, Bill Gates and key Microsoft strategists reviewed the design, which identified these primary requirements for the new OS:

- Architecture independence
- Multiprocessor support
- Multitasking and scalability
- Support for large memory space
- Distributed computing
- Compatibility with MS-DOS, 16-bit Windows, OS/2 and POSIX
- Government-certifiable security
- Highly-available
- Easy to administrate

Along with these requirements, the following design goals were defined:

- **Extensibility** - The capability to add new features and functions to the operating system without the need to modify the original software. In other words, allowing new features to be plugged in.
- **Architectural independence and support for SMP architectures** - The OS should be easy to moved from one processor architecture to another with as little re-coding as possible.
- **Reliability and robustness** - The OS should protect itself from error conditions, such as memory overlapping, but should also respond predictably to error conditions or even to hardware failure.
- **Performance** - The OS must be tight, fast, and as efficient as possible on all hardware platforms.
- **Compatibility** - Backwards compatibility with existing technology, such as MS-DOS and Windows must be maintained.

Windows NT 3.1 did a good job in living up to the defined criteria and was soon followed by version 3.5 and 3.51.

The consumer branch of Windows had evolved and, shortly after the release of Windows NT 3.51, Windows 95 was released. With its massive GUI change (compared to Windows 3.11, or Windows NT 3.51 for that matter) the once so transparent differences between the two were now becoming very apparent. The natural evolution of development was to incorporate the prior releases into one product: Windows NT 4.0.

One of the true assets of Windows NT 4.0 was that any non-skilled computer user who had used Windows 95 could sit in front of a Windows NT workstation and feel at home right away with a familiar GUI. Add to that Windows 95 and DOS emulation, NetWare and TCP/IP connectivity, Internet connectivity, the support for fast processors (including SMPs), and so on, one begins to appreciate why it became such a popular product.

Also, most MS-DOS, Windows 16-bit and Windows 32-bit applications were compatible with NT 4.0, which enabled users to migrate from Windows 95 to Windows NT 4.0 with minimal hassle.

Windows NT was originally developed on MIPS RISC-based systems so the developers would not stumble into the pitfall of including any non-portable Intel-optimized code and was eventually available in four processor architecture flavours: MIPS, Alpha AXP, Power PC and Intel.

With the introduction of Windows 2000 (originally Windows NT 5.0), the MIPS and Power PC architectures were dropped due to very low public interest. As the development of Windows 2000 progressed, the Beta 2 phase was the last build to support the Alpha AXP architecture. At the time of release in late 1999, Intel was the only supported architecture left.

Windows 2000, as opposed to Windows Me, is targeted for business use and can run on a standalone workstation or on a business network, with different packages to suit different businesses and hardware requirements.

Figure 9 on page 32 shows Microsoft's evolution from DOS to today's range of operating systems.

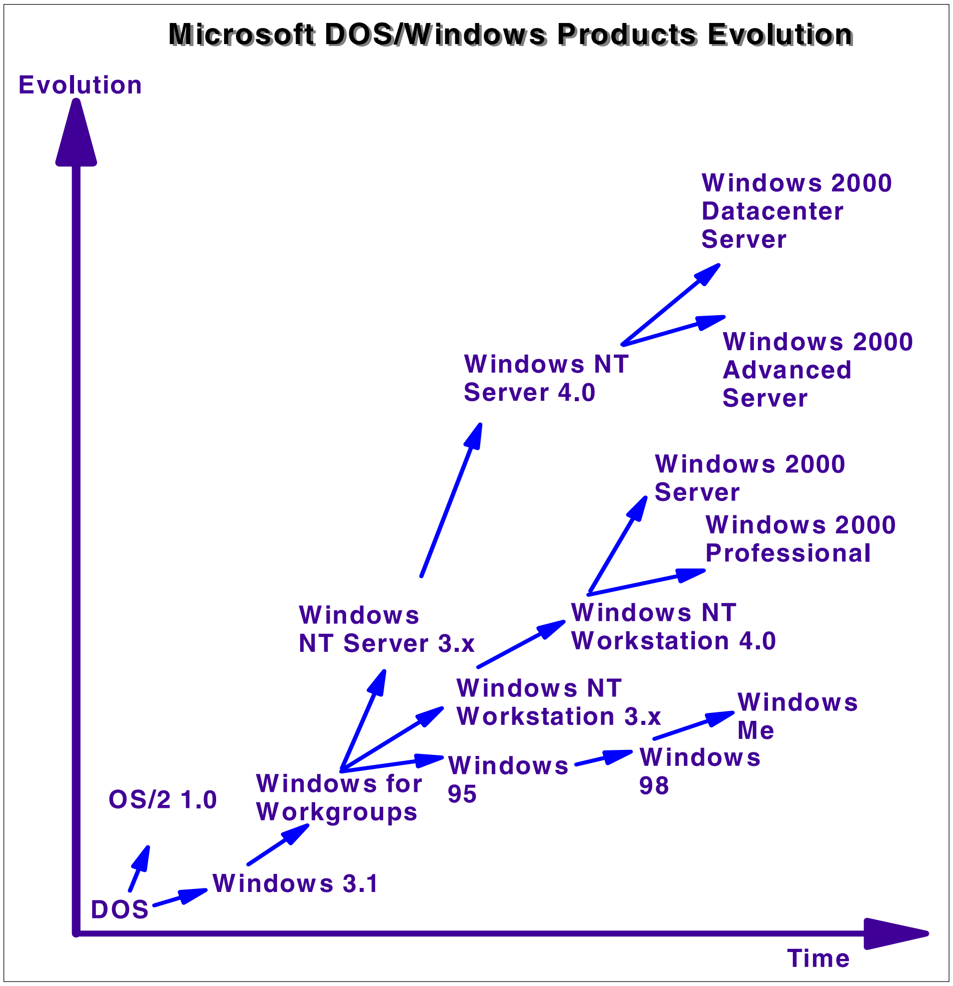

Microsoft DOS/Windows Products Evolution
Evolution
Windows 2000 Datacenter Server
Windows NT Server 4.0
Windows 2000 Advanced Server
Windows 2000 Server
Windows 2000 Professional
Windows NT Server 3.x
Windows NT Workstation 4.0
Windows NT Workstation 3.x
Windows Me
Windows 95
Windows 98
OS/2 1.0
Windows for Workgroups
Windows 3.1
DOS
Time

Figure 9. Microsoft DOS/Windows product evolution

Even though Windows 2000 is only available on the Intel platform, it has been made more capable of dealing with the ever-increasing performance of hardware on the market, with the Datacenter server designed to support 32-way symmetric multiprocessing (SMP), 64 GB of RAM and 4-node clustering.

Figure 10 on page 33 shows how Microsoft DOS, Windows, NT, and Windows 2000 products have evolved with increasing hardware performance.

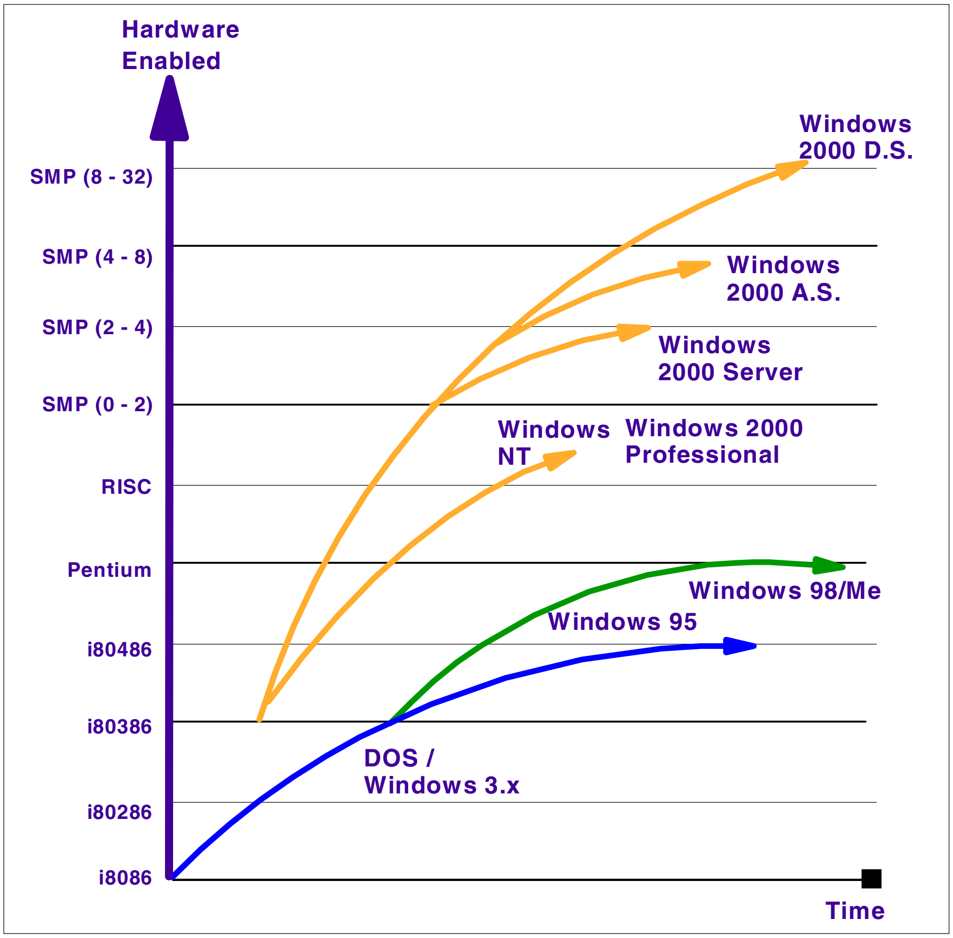


Figure 10. Microsoft operating systems, hardware enabled versus time

2.2.2 Windows 2000

The Windows 2000 operating system architecture is a hybrid architecture comprised of client/server, layered, object-oriented and symmetric multiprocessing architecture principles. Although it uses objects to represent shared system resources, Windows 2000 is not a strictly object-oriented operating system. The bulk of the code is written in C for both tool availability and portability and C does not directly support object-oriented constructs. However, Windows 2000 borrows from the features of object-oriented languages. Its features include 32-bit addressing, virtual memory support, preemptive multitasking, multithreading, multiprocessing support, system security and enhanced system integrity, platform independence and built-in modular networking. The diagram in Figure 11, taken from the Microsoft white paper “Windows 2000 Reliability and Availability Improvements”, illustrates the Windows 2000 Server architecture:

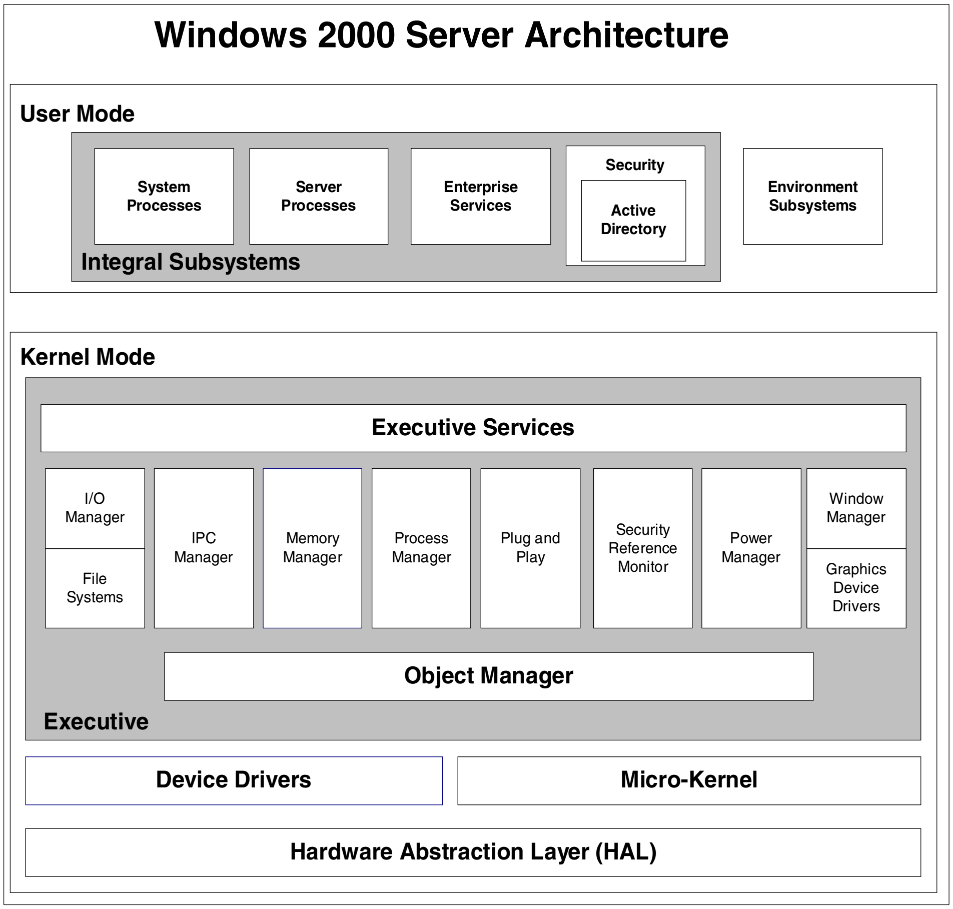


Figure 11. Windows 2000 reliability and availability improvements

A complete explanation of Windows 2000 architecture is beyond the scope of this Redbook; in fact, entire books on this subject have already been written. If you require an in-depth explanation we recommend the Windows 2000 Server Resource Kit. An online version of the Resource Kit can be found at:

`http://www.microsoft.com/windows2000/library/resources/reskit/default.asp`

This chapter attempts to explain Windows 2000 architecture from a high-level block diagram approach and will try to give an overview of Windows 2000 architecture without over-simplifying it.

The OS is built upon a layered approach similar to the UNIX operating system. One advantage of the layered operating structure is that each layer of code is only given access to the layer below (interfaces and data-structures). This structure also allows the OS to be debugged starting at the lowest layer and adding one layer at a time until the whole system works correctly. Layering also makes it easier to enhance the operating system; one entire layer can be replaced without affecting other parts of the OS.

Windows 2000 is a portable operating system because of two design decisions: First, the operating system was written in ANSI C, a language that enables programs to be ported easily to other hardware architectures. Second, all parts of Windows 2000 that must be written for a specific type of hardware are isolated in an area called the Hardware Abstraction Layer (HAL). To move Windows 2000 to a new hardware platform, developers need to do little more than recompile the C code for the new hardware and create a new HAL. Designing an OS around a HAL means that a large portion of the code is exactly the same between hardware platforms. This also means that only the small slice of code that interfaces with the computer's hardware needs to be rewritten as Windows 2000 is ported between different processor architectures, thus, providing a high level of portability.

There are two modes that Windows 2000 operates in: Kernel mode and user mode.

2.2.2.1 Kernel mode

In this mode, the software is able to access the hardware and system data as well as all other system resources. The kernel mode has the following components:

- **Executive** - This contains components that implement memory management, process and thread management, security, I/O, interprocess communication and other base OS services. For the most part, these components interact with one another in a modular, layered fashion.
- **Microkernel** - Its primary functions are to provide multiprocessor synchronization, thread and interrupt scheduling, and dispatching and trap handling and exception dispatching. During system startup, it extracts information from the registry, such as which device drivers to load and in what order.
- **Hardware Abstraction Layer (HAL)** - This is the code associated with Windows 2000 that changes with the hardware on which it is being run on, thus making it compatible with multiple processor platforms. The HAL manipulates the hardware directly.

- **Device drivers** - These are the kernel-mode code that interacts with hardware devices using a set of routines.
- **Windowing and graphics system** - This implements the Graphical User Interface (GUI).

2.2.2.2 User mode

Software in user mode cannot access hardware directly. The user mode-protected subsystem has four primary responsibilities:

- Special system support processes, such as the logon process and the session manager.
- Windows 2000 services that are server processes, such as the Event Log and Schedule services.
- Environment subsystems that provide an operating system environment by exposing the native OS services to user applications, namely: Win32, POSIX and OS/2 subsystems.
- User applications - Either Win32, Windows 3.1, MS-DOS, POSIX or OS/2.

User applications do not call the native Windows 2000 OS services directly; instead, they go through subsystem dynamic link libraries (DLLs). This translates a documented function into the appropriate Windows 2000 system service calls. The protected subsystems are divided into two groups, which are described in the following sections.

Environment subsystems

The environment subsystems are services that provide application programming interfaces (APIs) that are specific to the OS.

The three environment subsystems are the Win32, POSIX and OS/2 subsystems. Applications and subsystems form a client/server relationship in which the applications are the clients and the subsystems are the servers. One of the benefits of this type of architecture is that you can include support for other types of applications to Windows 2000 simply by adding subsystems.

Applications cannot interfere with each other since they are running in separate address spaces. OS code and data in the subsystems are protected from applications because subsystems also reside in their own address spaces. The Executive shares address space with running processes, but it is protected by the wall between kernel mode and user mode. It is impossible for an application to corrupt code or store data in the Executive because the processor notifies the operating system of invalid memory access before these things occur. You may have seen this as the infamous Blue Screen of

Death (BSOD). The reason a BSOD occurs is not really a computer crash but instead a protective measure from the OS before the crash. This way, data in memory can not be compromised by rouge drivers or other malfunctioning software.

Integral subsystems

The integral subsystems are services that provide the APIs for Win32 applications to call when they have to perform standardized operating system functions, such as creating windows and opening files. It has five main components, which utilize four main support functions:

- **Process and thread manager** - The process manager sees processes as objects. Its responsibility is to create and terminate processes and threads. It also suspends and resumes the execution of threads and stores and retrieves information about processes and threads.
- **Virtual memory manager** - The Virtual Memory Manager (VMM) performs three essential functions: Managing the virtual address space of each process, sharing memory between processes, and protecting each processes' virtual memory. It is also the underlying support for the cache manager. Each processor architecture implements virtual memory through hardware differently; therefore, the portion of Windows 2000 that directly interfaces with virtual memory hardware is not portable and must be re-coded when moving to another hardware architecture. Windows 2000 supports 4 GB of virtual memory and according to the Microsoft white paper "Windows 2000 Reliability and Availability Improvements:"

  "The upper 2 GB is reserved for kernel-mode processes and the lower 2 GB is shared by kernel-mode and user-mode processes." Figure 12 on page 38, also taken from the white paper, shows a graphical representation of the Virtual Memory Manager.

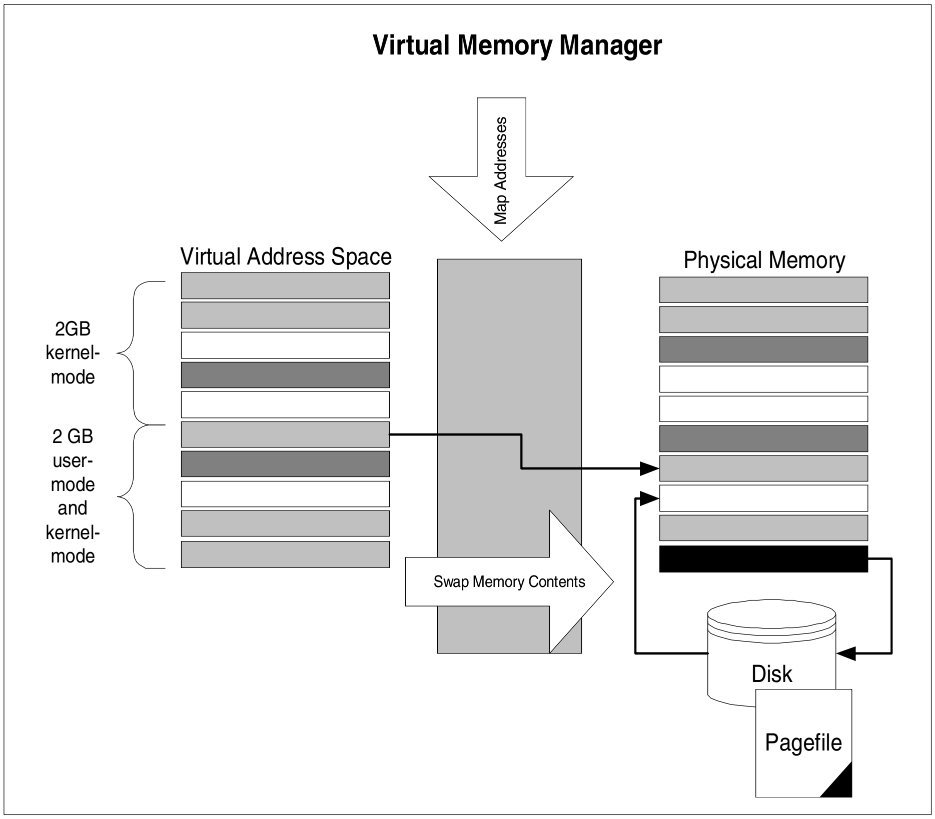


Figure 12. Windows 2000 Virtual Memory Manager

- **Security reference monitor** - The security reference monitor is responsible for controlling which objects have permissions to which resources. Each object has an Access Control List (ACL) that is queried when the object makes a service request. Access to resources is allowed or disallowed according to the rights the module has in the ACL.
- **I/O System manager** - The I/O manager is responsible for dispatching all system I/O requests. All I/O devices, network ports, printers, drives and so on are mapped to virtual files. These virtual files are referred to as file objects and are managed by the object manager just like any other object.
- **Cache manager** - The cache manager improves the performance of file-based I/O by causing recently-referenced disk data to reside in main memory for quick access. It also defers disk writing by holding the updates in memory for a short time before sending them to disk.

The support functions are:

- **Object manager** - The object manager creates, manages and deletes Executive objects. Executive objects are created in the Executive and are accessible to the Executive and protected subsystems. They can be thought of as message packets that represent things, such as processes, threads, semaphores and other low-level objects.
- **LPC facility** - Local Procedure Calls (LPCs) are used to pass messages between processes running on a single Windows 2000 system. Because LPC message passing requires quite a bit of overhead, the LPC facility is only utilized when an API must change global data. Otherwise, API routines can be implemented directly in a private Dynamic Link Library (DLL).
- **Run-time library functions** - Similar to string processing, arithmetic operations, data type conversion and security structure processing.
- **Executive support routines** - Similar to system memory allocation and interlocked memory access.

2.2.2.3 Multitasking and multithreading support

Windows 2000 is a Symmetric Multiprocessing (SMP) operating system. This means that Windows 2000 is designed to always execute its instructions on any available processor in the system. When there are more threads to run than processors to run them on (this is always the case on a single processor system), an SMP operating system also performs multitasking, dividing each processor's time among all waiting threads.

User-mode processes, as is the case with many other system resources, are defined as objects in Windows 2000. A process is much more than just a program; it is a data structure that is comprised of the following parts:

- An executable program that defines initial code and data
- A private address space that the VMM allocates for the process
- System resources that are allocated to the process
- At least one thread of execution

It is the goal of any multitasking system to make it appear to the user that it is accomplishing many tasks concurrently. Preemptive multitasking is accomplished in Windows 2000 by allowing threads to run only for a specified amount of time, called a time quantum. After a thread runs for the specified time quantum, it relinquishes control of the processor to the next waiting thread with the highest priority.

Threads are implemented in Windows 2000 in order to achieve this concurrency in a convenient and efficient way. Threads can be thought of as scaled-down processes; they require less overhead and are faster to create than processes.

A thread contains several components. In this discussion, the most important components are:

- A user-mode stack and a kernel-mode stack
- Register values representing the current state of the processor
- Private storage for use by subsystems, runtime libraries, and DLLs

These components make up what is called the thread’s context. Since a processor can only execute one thread at a time, it must switch between threads in order to multi-task the processor(s). This switching is called context switching. Basically, it involves running a thread for a specified amount of time, saving its context, loading another threads context, and then repeating this sequence for as long as there are threads waiting to execute. This switching is accomplished within the context of Windows 2000’s preemptive multitasking, as explained in the previous paragraphs.

2.2.2.4 Multiuser support

Architecturally, Windows 2000 is a single-user operating system. Multiple users can be defined, but only one user at a time can interactively log on.

The introduction of Terminal Services as a system service in Windows 2000 Server, as opposed to a new OS as Windows NT 4.0 Terminal Server Edition was, is a way of achieving a UNIX-like multiuser environment but until applications are written to fully support this environment, traditional single-user installations will still be the most common.

Users in Windows 2000 have certain security privileges or rights to resources on the system. When Windows 2000 is installed, a user, called Administrator (which is similar to the root user in AIX), is created. The Administrator has full control of all files and resources on the system. He or she may also create other users and give them lesser rights to resources on the system.

When connected to a network, Windows 2000 systems can interoperate within a domain or in a peer-to-peer relationship (called a workgroup). In a workgroup model, each computer maintains its own user account database. Each user logs on to a local account on their workstation. Computers can share resources over the network by using the concept of shares.

In a domain model, a Windows 2000 Server maintains a centralized user account database. Users log on to the domain regardless of what computer they are using or they can select to log on locally to any computer that is part of the domain, providing there is a local user account database present.

Chapter 3. Open System Standards compliance

At a very high level, Open System Standards are a means of ensuring portability of both applications and users, and a means of ensuring inter operability.

Application portability to any platform guarantees independence from the platform provider. User portability minimizes education and training through consistent user interfaces. All these features can be categorized as customer investment protection. Also, providers/suppliers can expect an even larger market for their products due to the portability of their applications.

In the case of inter operability, Open System Standards become even more important. Customers that choose to purchase a product that conforms to Open Systems Standards can be reasonably assured that products from other providers will work with that product in a manner consistent with any other vendor that complies with the same standards.

3.1 Open standards groups/bodies/committees

The objective is for the leaders of a particular industry to get together with leaders from academia and other groups and define a set of rules, guidelines, and definitions for a systematic approach to portability and inter operability that suppliers agree to abide by.

These standards groups are also responsible for certifying that a product complies with a particular standard. There is also a forum to include comments from all interested parties at large. This is known as the Request For Comments (RFC) process. The very act of participation by a wide range of people and groups is why these adopted standards are referred to as Open System Standards.

3.2 The Open Group example

The Open Group is a vendor-neutral international consortium of more than 200 members including leaders in government, academia, world-wide finance, health care, commerce and telecommunications, that has a combined IT budget in excess of $55 billion annually. The Open Group's breakthrough IT DialTone initiative will ensure that the Internet remains open by collecting a core set of specifications, products, and technologies to create a common level of global security and reliability. The Open Group charter provides for research and development of non-proprietary technologies and specifications through collaborative development as well as a unique testing and branding process for guaranteed compatibility of commercial products.

The Open Group was formed in February 1996 by merging X/Open Company Ltd. and the Open Software Foundation. The nine sponsors of The Open Group are Digital Equipment Corporation, Hewlett-Packard Company, Siemens Nixdorf Information system AG, Fujitsu Limited, Inc., Hitachi Limited, International Business Machines Corporation, NCR Corporation, Novel, Inc., and Sun Microsystems, Inc. The Open Group can be reached online at:

`http://www.opengroup.org/`

3.3 AIX Open Systems Standards compliance

There are a number of organizations that have a major impact on the direction of AIX. The AIX development organization is making a commitment of time and resources to support these groups and their efforts. Some of those groups are:

- **IEEE (Institute of Electrical and Electronics Engineers)** - IEEE is an accredited standards organization. POSIX is a group of related standards that are being developed to support application portability.
- **Open Group (Open Group Limited)** - Founded in 1984, Open Group ®, formerly known as X/Open®, is a worldwide, independent company dedicated to bringing the benefits of open computer systems to market. Open Group produces specifications in support of application portability and interoperability. A branding program is in place that allows compliant products to obtain an Open Group brand.
- **OMG (Object Management Group)** - The Object Management Group is defining specifications to support application portability and interoperability in a distributed object oriented environment.
- **Open Blueprint** - The Open Blueprint is an IBM standards-based framework for a unified, open, distributed client/server environment. It is intended as a guide for IBM and other developers as they provide products and solutions that interoperate and integrate with other installed products.

AIX 5L conforms to several industry standards for compatibility with other UNIX systems based on review by established standards organizations. These include support for a range of specifications relating to items, such as:

- Product standards
- System calls and libraries

- Operating system interfaces and environments (API)
- Application portability at the source-code level (compilers)
- Commands and utilities
- TCP/IP interoperability
- Portability of users and programmers
- Distributed network-transparent window system
- Distributed file systems
- Communication protocols
- Magnetic media
- System security
- System administration

Here is a partial list of the standards that help define AIX 5L:

- X/Open UNIX 98 Product Standard
- ANSI/IEEE 1003.1:1996 and 1003.2:1992
- XPG4 NFS and XPG5
- FIPS PUB 151-2 and 189
- BSD 4.3 Compatibility

Linux affinity on AIX includes Linux application source compatibility, and compliance with emerging Linux standards.

For a more detailed and latest informations about AIX standards compliance, see the online reference at:

`http://www.austin.ibm.com/software/Standards/`

3.4 Windows 2000 Open Systems Standard compliance

Microsoft's interoperability strategy is based on a four level framework that covers network, data, applications, and management integration.

Microsoft supports virtually all of the common TCP/IP network interoperability standards, such as SNMP, FTP, HTTP, and telnet. There are also add-on products available for UNIX interoperability and services for Novell Netware.

DNS integration is based on RFC 2052 and RFC 2136 making it BIND 8.2.2 compatible.

For data interoperability, Windows 2000 uses Open Database Connectivity (ODBC) allowing application developers to access data irrespective of the database or the operating system it is running on.

Management interoperability is achieved with SNMP-based system management.

Perhaps more important to Microsoft than Open Systems Standards compliance is user interoperability. Windows 2000 is aimed at a range of machines from the standalone desktop to an SMP machine capable of running a business. The similar look and feel from Windows 98 and Windows NT to the new Windows 2000 is quite important. Microsoft is more likely to be concerned about this portability and the *de facto* standards it has set than the Open System Standards.

For more information on Windows 2000 and its interoperability capabilities, go to:

```
http://www.microsoft.com/Windows/server/Overview/features/interop.asp
```

Microsoft has defined standards of their own which partners and independent software vendors (ISVs) are required to meet for Windows 2000 logo certification. Details of these requirements can be found online at:

```
http://msdn.microsoft.com/certification/appspec.asp
```

Information on the "Designed for Windows" Logo Program is also available online at:

```
http://msdn.microsoft.com/winlogo/
```

Chapter 4. Packaging

The way an operating system is packaged may not seem the most important aspect when comparing several operating systems. However, it becomes very important when you start looking at which package/products you must buy to achieve a specific set of functions and support your hardware requirements. Questions, such as what functions are included in the base operating system and how many users are supported for a license, need to be clarified before a final decision can be made.

This chapter discusses how AIX Version 5L and Windows 2000 are packaged and what it is the customer actually receives. While AIX Version 5L offers one package that can run anything from a low-end graphics workstation to the high end parallel SP system, Windows 2000 offers four packages ranging from the Professional version, designed for the workgroup computer, to the Datacenter version that is designed for ISPs and can support up to 16 processors.

4.1 AIX version 5L packaging

AIX uses a flexible packaging scheme. Behind the scenes, AIX is actually comprised of many file sets. A file set is the smallest installable unit of AIX. The notion of file sets comes from the Open Group's specification for product packaging. This permits the customer to only install the filesets that are needed to accomplish the particular mission of the system. While this can be difficult to understand for a new AIX administrator, it is critically important to an experienced system administrator who wants to:

- Craft or tailor a system to a specific mission
- Use only as much disk space as is needed
- Gain maximum system performance

For those who are new to AIX administration, AIX 5L has pre configured offerings that are tuned for a generalized operating environment and installed with the minimum amount of administrator intervention straight out of the box. It is also possible to have AIX pre installed on a new RS/6000 or IBM @server pSeries servers. These, along with the manuals and publications included with any AIX license, simplify the installation process.

The base install (whether done before, as a pre-install, or after the RS/6000 or IBM @server pSeries server arrives) will only configure the device drivers for those devices that are connected at the time of installation. Others can be easily added at a later date using either SMIT (as explained in section 8.1.4.2, “SMIT” on page 244) or Web-based System Manager (explained in detail in section 8.1.4.1, “Web-based System Manager” on page 232). Once the base install has been completed, you can opt to install filesets individually or choose from any or all of the following five bundles that AIX makes available:

- Server
- Client
- Application development
- Personal productivity
- Media defined

Previous versions of AIX used the same packaging methods that Windows 2000 currently employs and had separate packages available for clients and servers. To simplify the matter, there is now one AIX package that can run any machine, and the pricing is completely dependent on the number of licensed users that are required; a base licence is one price, as is each individual user, no matter what system it is going on.

4.1.1 Package

AIX 5L supports POWER-based platforms (RS/6000 and IBM @server pSeries server) and Itanium-based platform. So there are two types of AIX 5L packages for each platform.

4.1.1.1 AIX 5L package for POWER-based platform

The single AIX package that is available to run on all RS/6000s and IBM @server pSeries servers is broken down into the following components:

- AIX 5.1 POWER Base Media
- AIX 5.1 POWER Expansion Pack
- AIX 5.1 POWER Bonus Pack
- AIX 5.1 Pubs/Documentation CDs

Along with the installation CDs, which include softcopy manuals, there are some hardcopy publications included to get you started. These publications are specific to AIX 5L Version 5.1 and include some that are even more specific to the modification level you have received. The publications that are generally received are as follows:

- *AIX Quick Installation and Startup Guide*, SC23-4111

- *AIX 5L Version 5.1 Installation Guide*, SC23-4112
- *AIX 5L Version 5.1 Network Installation Management Guide and Reference*, SC23-4113
- *AIX 5L Version 5.1 Quick Beginnings*, SC23-4114
- AIX 5L Version 5.1 Release Notes
- AIX 5L Version 5.1 Bonus Pack Release Notes
- Plus the license agreements and release notes for some separate items in the bonus pack

AIX is only available on CD-ROM; this is what will be shipped with AIX regardless of whether it is running on a standalone graphics workstation, a 24-way S85 system, or an RS/6000 SP (Scalable POWERparallel) system.

4.1.1.2 AIX 5L package for Itanium-based platform

The single AIX package that is available to run on Itanium-based platform is broken down into the following components:

- AIX 5.1 IA-64 Base Media
- AIX 5.1 IA-64 Expansion Pack
- AIX 5.1 Pubs/Documentation CDs
- AIX 5.1 IA-64 Supplemental Device Driver CDs

4.1.2 AIX 5L Expansion Pack and Bonus Pack

Expansion Pack is a vehicle for the delivery of new IBM and non-IBM products. AIX 5L Expansion Packs complement the operating system by providing encryption support, a browser to view on-line publications, and an HTTP server to serve on-line publication pages and support Web-based System Manager. One Expansion Pack is included with every shipment of AIX 5L when media is selected. The AIX Bonus Pack is a collection of extra software included at no additional charge with purchases of AIX Version 5L licenses when media is selected. It includes software from IBM and third-party providers.

POWER Expansion Pack

POWER edition includes:

- Network Authentication Service Version 1.1
- AIX Certificate and Security Support Version 4.0
- AIX Certificate and Security Support Version 5.0
- Data Encryption Standard Library Routines Version 5.1

- IPSEC Encryption
- LDAP 3.2 Encryption
- System V Utilities
- Web-based System Manager Security Version 5.0
- Netscape Communicator 4.76
- IBM HTTP Server Version 1.3.12

Itanium Expansion Pack
IA-64 Expansion Pack contains:

- AIX Certificate and Security Support Version 4.0
- Data Encryption Standard Library Routines Version 5.1
- IBM HTTP Server Version 1.3.15
- IPSEC Encryption
- LDAP 3.2 Encryption
- System V Utilities
- Web-based System Manager Security Version 5.0
- Netscape Communicator 4.76

POWER Bonus Pack
POWER Expansion Pack, shipped with AIX Version 5L contains:

- Fast Connect Try and Buy Version 3.1
- SCO Tarentella 1.4

4.1.3 Options

In addition to AIX 5L, there are a number of options that can be selected to extend the packages already available to you.

4.1.3.1 Infoexplorer

Infoexplorer is the updated Hypertext Libraries. This was used in previous versions of AIX to view the softcopy manuals. Since all the manuals for AIX 5L are in HTML format, this product is not required unless Infoexplorer is required for other software product manuals.

4.1.3.2 AIX Fast Connect

AIX Fast Connect is available as an option for either Windows or OS/2. AIX Fast Connect for Windows Version 3.1 enables AIX to be a part of the Microsoft Network Neighborhood. Windows clients have AIX file and print capabilities without the need for additional software on the clients. Using Microsoft Common Internet File System (CIFS) and server message block (SMB) protocols over TCP/IP, Windows clients can access:

- AIX journaled file system (JFS).
- CD file system (CDFS).
- Network file system (NFS) mounted subsystems.
- AIX printing services.

AIX Fast Connect for Windows includes:

- Support for Windows for Workgroups (WFW), Windows 95, Windows 98, Windows NT 4.0, and Windows 2000 clients.
- CIFS client resource browsing, which enables users to view available shared AIX files and printing services.
- Support for CIFS long file names.
- NetBIOS request for comments (RFC) 1001/1002 allows programs written for a NetBIOS environment to run over TCP/IP. NetBIOS is only for use with the Fast Connect features and is not available for other applications on AIX.
- Use of the TCP/IP domain name system (DNS) to resolve NetBIOS machine names.
- Microsoft WINS server support.
- Opportunistic locking.
- Unicode representation of share, user, file, and directory names.
- AIX authentication/authorization with encrypted passwords using 56-bit encryption.
- DCE/DFS integration.
- Interaction with NT Server Version 4.0 to:
  - Provide PC user authentication and authorization.
  - Work with the NT Server domain master browser to find and publish shared resources across TCP/IP subnets.
  - Work with the NT Server Windows Name Server (WINS) to resolve NetBIOS machine names.
- Support for JFS-ACLs.
- HACMP support, using server name and aliases.

- Command-line (NET command subset), SMIT, and Web-based System Manager configuration.
- HTML-based publications (English only).
- IBM service and support.

AIX Fast Connect for Windows provides at least twice the performance of its predecessor, AIX Connections. AIX Fast Connect uses the TCP/IP sendfile API with an in-kernel network file cache to improve network TCP/IP performance. AIX Fast Connect for Windows also assumes the presence of at least one Windows 2000 server in the network environment to provide Microsoft Master Browser support.

With the latest release of AIX Fast Connect, Version 3.1, there have been the following enhancements:

- Locking enhancements:

  Some applications require shared files between AIX server-based applications and PC client applications. The file server requires lock mechanisms to protect these files against multiple modifications at the same time. Because of this, Fast Connect implements UNIX locking, in addition to internal locking, to allow exclusions based on file locks taken by PC clients.

- Per share options:

  Several advanced features of AIX Fast Connect are available as per-share options. These options are encoded as bit fields within the sh_options parameter of each share definition. These options must be defined when the share is created with the `net share /add` command, or set through the SMIT file share panel.

- PC user name to AIX user name mapping:

  When a client tries to access resources on the server, it needs to establish an SMB/CIFS session. SMB/CIFS session setup can use user level security or share level security. In case of user level security, clients must present their user names. In previous Fast Connect releases, it was required that the user name match the one on AIX exactly. In many situations, this one-to-one mapping of user name is not possible. AIX Fast Connect on AIX 5L allows the server administrators to configure the mapping of PC user names to AIX user names, and then uses that server user name for further user authentication and AIX credentials.

- Windows Terminal Services support:

Windows Terminal Services allow support for multiple users on one Windows 2000 machine. When a multiuser Windows 2000 machine connects to a Fast Connect server for File and Print Services, Fast Connect allows multiple SMB sessions over one transport session. In previous releases, Fast Connect was limited to one SMB session per transport connection.

- Search caching

  Generally, file search operation requests from a PC client takes large amounts of resources. Additionally, performance issues may arise if a large number of clients do file search operations at the same time. AIX Fast Connect allows the user to enable search caching. If enabled, all the cached structures will compare their time stamps to the original files to check for modifications periodically. This feature improves file searching significantly.

4.1.4 National Language Support

All AIX operating system packages can support over 60 languages. The language support is shipped on the installation media. Customers do not have to purchase separate media to get a particular language.

A unique aspect of the AIX language support is that multiple languages, or locales, can be supported at the same time. Many operating systems allow only one language at a time, and it must be the same for all users of the system.

However, an AIX user can interact with the system in one language in one window and, concurrently, with another language in another window. Also, different users of the same system can use different languages.

To accomplish this, the system administrator must load the appropriate language support from the installation media, and the user must specify which language to use by way of an environment variable called LANG.

AIX 5L is also EuroReady. This means that when used in accordance with its associated documentation, AIX is capable of correctly processing monetary data in the euro denomination and respecting the euro currency formatting conventions (including the euro sign). This is assuming that all other products used with AIX (including hardware) are also EuroReady.

Information on what languages are supported can be found in section 8.1.10, “AIX support for national languages” on page 280.

4.1.5 Bus support

The IBM @server pSeries systems have, over the years, been either microchannel, PCI and ISA, or just PCI. AIX Version 5L for POWER supports the microchannel, PCI and ISA buses.

The Micro Channel Architecture (MCA) is a 32-bit bus designed by IBM that, for many years, was the standard bus in the RS/6000s. This was an IBM proprietary architecture and has been phased out of the product range.

The PCI/ISA systems represent a major paradigm shift for the RS/6000 and IBM @server pSeries server product line. The Web Server division is moving from a line of specialized machines with custom-built components in fairly low volumes to a more generalized line of systems with common off-the-shelf components. There are advantages here for both the user and IBM. IBM can now design new systems and release them to the market faster than their competition. In addition, because of the use of fairly common high-volume parts from the Personal Computer line, such as power supplies, cables, disks, memory, and so on, prices continue to drop for the user. The PCI slots in the new machines are a mixture of 32-bit and 64-bit.

The recent IBM @server pSeries systems that have been released no longer have the combined PCI/ISA bus. These models are purely PCI and have a combination of both 32- and 64-bit buses.

AIX 5L supports all of these so that it can run on any machine in the RS/6000 range including those that are no longer sold. At this writing, the universal serial bus (USB) is not supported on AIX 5L Version 5.1. This is currently being developed for future releases.

4.1.6 Licensed Program Products (LPPs)

Any operating system will normally require additional software or middleware to accomplish specific tasks. While AIX is a very robust and utilitarian operating environment, specialized products are often needed to produce a complete solution for a particular end-user requirement. Applications, such as databases, graphics-rendering products, and personal-productivity applications, are available from IBM and other manufacturers. These additional software products, which can be ordered separately, are referred to as Licensed Program Products (LPPs).

The AIX Application Availability Guide is a list of IBM software that is available to run on AIX. This list provides information on which versions of AIX the software is compatible with, as well as when the software will be withdrawn from the market (and support) and what the replacement product will be. There is also a guide for withdrawn software should you need to trace an upgrade path for existing products. These guides are available online at:

```
http://www.ibm.com/servers/aix/products/ibmsw/list/
```

4.2 Windows 2000 packaging

Windows NT Version 4.0 originally had two packages to offer: A workstation and a server. This was later expanded to include the Enterprise Edition as well as the Terminal Server Edition. Windows 2000 packaging has a slightly different approach and comes in the following four packages:

- Professional
- Server (also known as Standard)
- Advanced Server
- Datacenter Server

These four packages are designed to cater to the increasing number of Intel-based systems that are increasing their SMP (symmetric multiprocessing) performance. While the Professional, Server, and Advanced packages of Windows 2000 are being sold as individual products, the Datacenter Server is only sold as a complete solution, including certified and tested hardware.

In most situations, upgrading to Windows 2000 will mean getting a box with the CD and a basic installation manual (as well as the license information). This will not differ for a new license on a new machine where it will be pre-installed, only the basic information required for installation is shipped with the box.

When you install any of the Windows 2000 Server packages, you must specify whether the product is to be licensed in Per Seat or Per Server mode. Microsoft suggests the following guidelines for selecting the licensing mode:

- If the machine is to be the only Windows 2000 server in your network, select the Per Server option, and specify the number of client access licenses (CALs) purchased for the product. This sets the maximum number of concurrent connections that can be made to the server. As you purchase additional licenses, you can increase this number.
- If multiple Windows 2000 servers will be run in your network and the total number of CALs required to support Per Server mode for all the servers is equal to or greater than the number of client computers or

workstations, select the per seat option for the product on all computers in your network.

Windows 2000 will only install device drivers for the detected hardware but, as the majority of all drivers reside in one 50MB big CAB-file gets copied to your hard drive (%SystemRoot%\Driver Cache\i386\driver.cab) during installation, adding hardware at a later time rarely requires the Windows 2000 CD.

At the core of all Windows 2000 offerings is the Active Directory structure. It is designed to simplify management, strengthen security and extend interoperability. More detailed information is available in Section 10.3.18, “Active Directory” on page 487.

4.2.1 Windows 2000 Professional

Windows 2000 Professional is considered a professional OS, is designed for business usage, and is the upgrade from (or replacement for) Windows NT Workstation Version 4.0. It supports a maximum of two CPUs and 4 GB of physical memory. It was designed to incorporate the best business features of Windows 98 and to build on the strengths of NT, thus improving security and reliability.

4.2.2 Windows 2000 Server

Windows 2000 Server is meant for departmental file and print, web and entry-level application servers, and is the upgrade from Windows NT Server Version 4.0. It supports up to 4-way SMP and 4 GB of physical memory and is aimed at small to medium enterprise application deployments and organizations with numerous workgroups and branch offices. Windows 2000 Server has the following features:

- Windows management tools
- Kerberos and Public Key Infrastructure (PKI) Security
- Windows Terminal Services
- Enhanced Internet and Web services

4.2.3 Windows 2000 Advanced Server

This is the replacement for Windows NT Server 4.0, Enterprise Edition. It is aimed at providing a comprehensive clustering infrastructure for high availability and scalability. It supports up to 8-way SMP and 8 GB of physical memory. It is designed for database-intensive work and has a load-balancing component for system and application availability. The Windows 2000

Advanced Server builds on the existing features of the Standard package and also has the following features:

- Enhanced application fail-over clustering
- High-performance sort
- Four-node Cluster Services
- Network Load Balancing

4.2.4 Windows 2000 Datacenter Server

A new area for Windows 2000 is the Datacenter Server Package. It supports 32-way SMP and 64 GB of physical memory. It has the same high availability and clustering that the Advanced Server edition advertises and is aimed at large data warehouses, science and engineering simulations, economic analysis, OLTP (online transaction processing) and large scale ISPs. The Windows 2000 Datacenter package builds on the existing features of the Advanced offering and offers more advanced clustering.

The minimum system requirements for a Datacenter solution are:

- 8-way SMP or higher server (supports up to 32-way)
- Pentium III Xeon processors or higher
- 256 MB of RAM recommended
- 2 GB hard disk with a minimum of 1 GB free space
- CD-ROM or DVD drive
- VGA or higher resolution monitor

As mentioned earlier in this chapter, it is not possible to buy the OS without accompanying hardware.

4.2.5 National Language Support

Multilingual editing and viewing features are standard in all Windows 2000 versions allowing users to view and edit information in over 60 languages.

By installing a Windows 2000 Multi-language module, users can change the language of the OS user interface. This allows a user to log on to a workstation and use any of the 24 languages in Table 1 on page 58.

Table 1. Supported Windows 2000 languages

Language	Professional	Server	Adv. Server	Multi Language
Arabic	X			X
Brazilian	X	X		X
Chinese (Simplified)	X	X	X	X
Chinese (Traditional)	X	X	X	X
Czech	X	X		X
Danish	X			X
Dutch	X	X		X
English	X	X	X	X
Finnish	X			X
French	X	X	X	X
German	X	X	X	X
Greek	X			X
Hebrew	X			X
Hungarian	X	X		X
Italian	X	X		X
Japanese	X	X	X	X
Korean	X	X	X	X
Norwegian	X			X
Polish	X	X		X
Portuguese	X	X		X
Russian	X	X		X
Spanish	X	X	X	X
Swedish	X	X		X
Turkish	X	X		X

The language modules can either be pre-installed on the machine by the administrator or published as applications using Intellimirror and be installed on demand by the users.

You will find more information in section 8.2.10, “Windows 2000 support for national languages” on page 326.

4.2.6 Bus support

Windows 2000 continues to support the PCI, ISA, EISA, PCMCIA and CardBus buses, although, Windows 2000 Remote Installation Services may not fully support unattended installations on computers that contain ISA devices or those that are not Plug and Play aware. The Universal Serial Bus (USB) and FireWire (IEEE-1394) are also supported. Windows 2000 has dropped support for the Microchannel bus.

4.3 AIX 5L and Windows 2000 positioning

Both AIX and the various Windows 2000 packages are designed to cover a large range of machines, from individual workstation scaling to high-end SMP servers. And, in the case of AIX, the highly-scalable SP systems.

Figure 13 on page 60 shows the range covered by these packages.

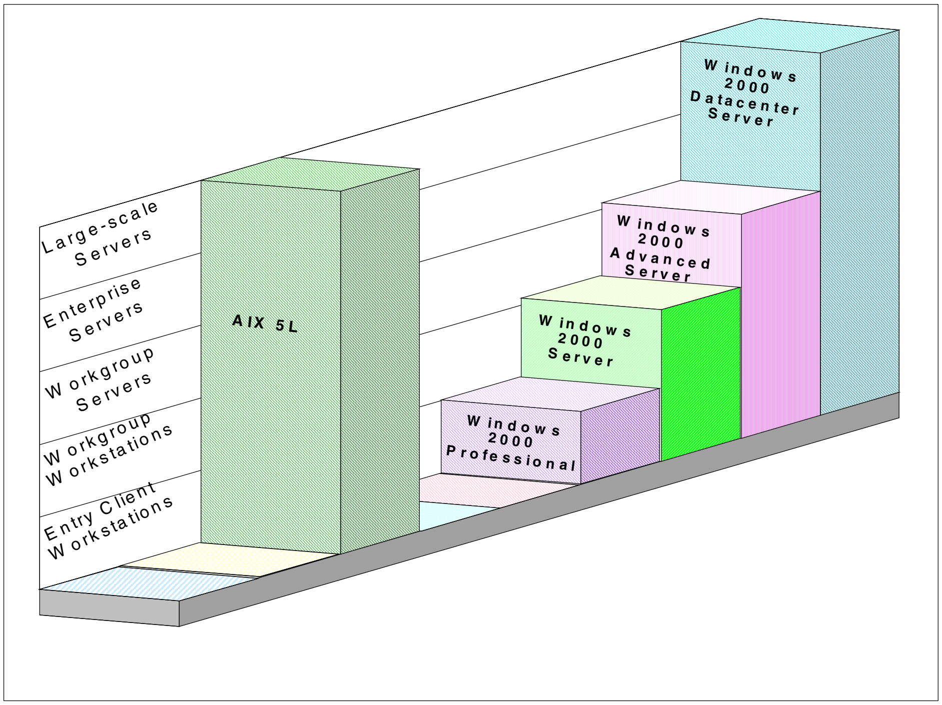


Figure 13. AIX 5L and Windows 2000 positioning

The September 1999 release of AIX Version 4.3.3 is the first release of AIX to be influenced by Project Monterey. Project Monterey is an initiative led by IBM with SCO, Sequent, and Intel to deliver a single UNIX product line that spans IBM @server pSeries systems, IA-32, and the upcoming IA-64 based systems. You can find more information on Project Monterey online at:

`http://www.ibm.com/servers/monterey/`

or

`http://www.projectmonterey.com/`

Chapter 5. User interface/graphics support

This chapter outlines the main features of the user interfaces and graphics support of both AIX and Windows 2000. This chapter does not intend to advocate the use of either of these products or to make a direct comparison between the two. The intention is to present readers with a neutral overview of both products and allow them to make their own decision as to the merits of each.

5.1 AIX Version 5L user interface

The default graphical user interface (GUI) that is provided with AIX is the Common Desktop Environment (CDE 1.0). CDE was jointly developed by IBM, HP, SUN, and Novell to provide a standard user interface across all UNIX flavors and, thereby, create truly open systems. CDE is made up of various components that interact with each other to form the desktop environment. These components are:

- Login Manager
- Session Manager
- Front Panel
- Style Manager
- Workspace Manager
- File Manager
- Application Manager
- Help Manager

In the following sections, we will take a closer look at the individual components and how they fit together.

5.1.1 Login Manager

The Login Manager provides a graphical login screen and manages user access to the system. The Login Manager is responsible for the following tasks:

- Reading initial configuration files
- Starting the X server for local displays
- Displaying the login screen and validating user login IDs and passwords
- Invoking the CDE Session Manager

The graphical user interface presented by the Login Manager and shown in Figure 14 can be fully customized by the system administrator. See the IBM Redbook *RS/6000 Graphics Handbook*, SG24-5130, for more details.

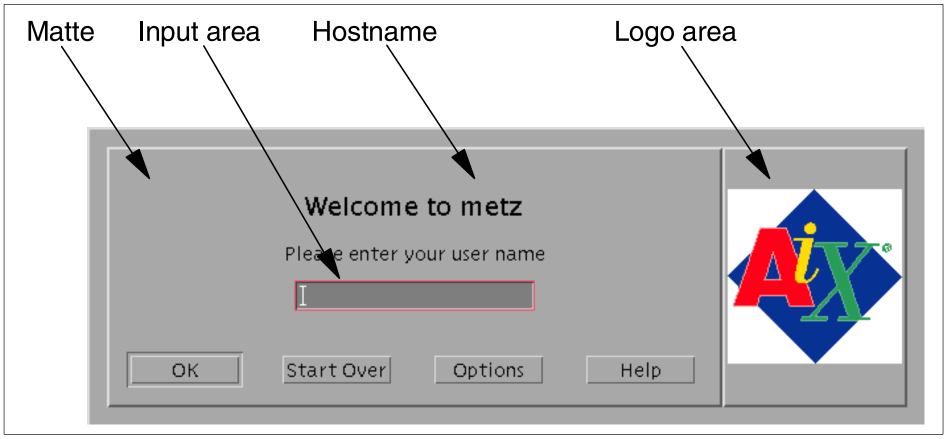


Figure 14. Login Manager user interface

An important feature of the CDE Login Manager is that it uses a Pluggable Authentication Modules (PAM) architecture. This enables the system administrator to use additional authentication methods to authenticate the user login and, thereby, provide enhanced security.

If necessary, it is possible to disable the Login Manager and use the standard AIX command line login. If the Login Manager has been disabled, the CDE desktop can be started manually from the command line by running the /etc/rc.dt shell script.

Once the user has terminated the CDE session, they will be returned to the login manager or the command line depending on the chosen implementation.

5.1.2 Session Manager

The Session Manager is the desktop software component responsible for initializing each user's desktop session after login. The Session Manager will automatically configure the desktop to reflect either the way the desktop was the last time the user logged out or a configuration that was previously saved by the user. These two automatic configuration possibilities are called the current session and the home session. These sessions are, in essence, snapshots of the desktop when the session was saved, and they contain information about:

- Applications running at the time the session was saved.

- Any desktop customization changes that were made by the user.
- The values of all of the X resources known to the X resource manager.

Each time a user logs out of CDE, the Session Manager takes a snapshot of the desktop just before the user logs off. This snapshot is saved as the current session and can be used the next time the user logs in to re-create the same desktop. Similarly, the user can, at any time, request that the Session Manager take a snapshot of the desktop and store it as the home session.

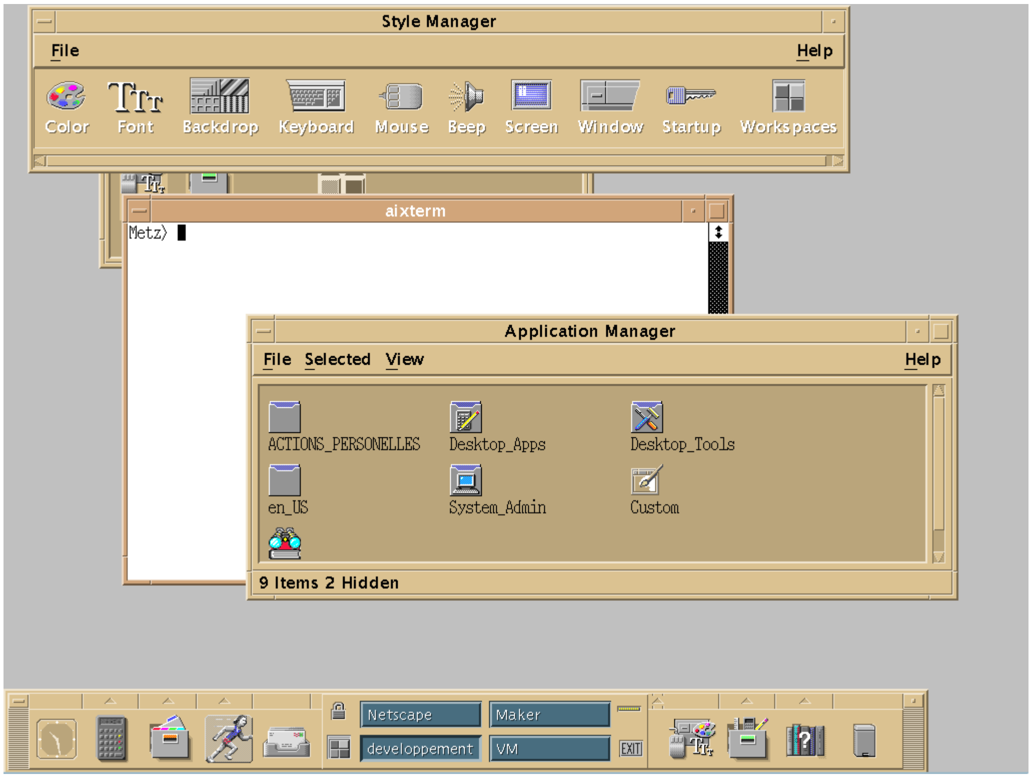


Figure 15. A sample CDE session

The CDE Session Manager is Inter-Client Conventions Manual (ICCM) compliant. This means that if an ICCM-compliant application is running in the desktop, not only will the session manager restart this application when the user logs back in, but the application will be started where the user left off. This means that the Session Manager is able to save the internal state of the application.

5.1.3 Front Panel

The Front Panel is the main desktop interface. As with all the components of CDE, it can be fully customized to meet the user's requirements. The default Front Panel, pictured in Figure 16, includes (from left to right):

- A clock
- Access to the desktop calendar program
- Access to the File Manager
- Access to personal applications
- Access to the desktop mail program
- A display lock
- Access to the Graphical Workspace Manager
- Access to the various workspaces
- A busy indicator
- An exit button
- Access to the printer queues and a printer information application
- Access to the Style Manager
- Access to the Application Manager
- Access to the Help Manager
- A trash can

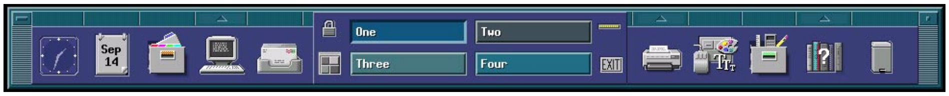


Figure 16. Default front panel

5.1.3.1 Front panel components

The Front Panel is made up of several different types of control components and four types of container components that group the controls. The available control components are:

- **Icon** - The icon control is the most common type of control used in the Front Panel. This type of control is actually a desktop object embedded into the Front Panel. (See section 5.1.6.1, "Objects" on page 71, for an explanation of desktop objects.) Each icon control is represented by the object's icon in the Front Panel. When an icon control is activated by either a single mouse click on the icon or by dropping another object onto it, the underlying object’s action that corresponds with the activation method used (mouse click or dropped object) will be invoked. The icon of an object used in the Front Panel can either be static or animated, which changes and animates when you select it. The icon control also has facilities that extend the functionality of an object. These include the capability to have the object’s icon change when an identified file is created or when mail arrives. Additionally, the icon control can be set up to allow only one instance of the object’s action to be active at any given time.The controls are:

- **Busy** - This control blinks to indicate that the Front Panel is actively processing your selection or drop request. This control only blinks when the Front Panel is processing a request. After the Front Panel finishes the request, the indicator will stop blinking.
- **Client** - This type of control allows you to display the graphics of an X Window program directly in the Front Panel. Obviously, the X Window client should produce graphic output that is appropriately sized for display in the Front Panel.
- **Clock** - This control is the clock in the default Front Panel.
- **Date** - This control is the calendar in the default Front Panel. When selected, this control will bring up the desktop calendar program.
- **File** - This type of control is used to represent a data file in the Front Panel. When a file control is selected, the action associated with the control is executed and is passed the file name of the file represented by the control.

There are four types of container components that are used to hold these controls in the Front Panel:

- **Main Panel** - The main panel, shown in Figure 17 on page 66, is the outermost container of the Front Panel. It holds the other containers and has the following special controls of its own:
  - Menu button
  - Iconify button
  - Position handle

  There can only be one main panel container in the Front Panel.

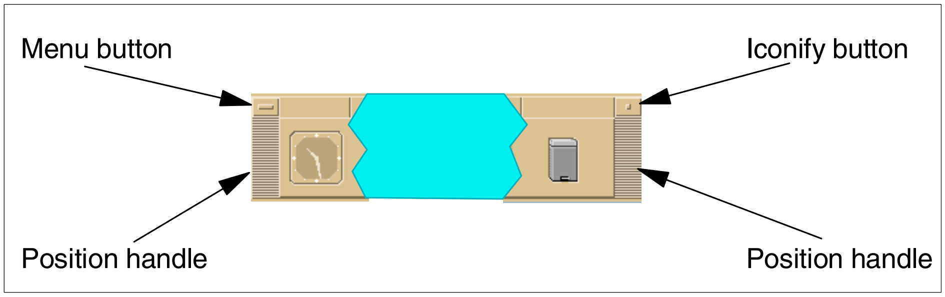


Figure 17. Main panel

- **Boxes** - Boxes are containers that reside in the main panel container and actually hold the controls that make up the Front Panel. The default Front Panel contains one box that holds all of the controls (except for the above-mentioned controls in the main panel container) in the Front Panel. Boxes lay out their contents in a horizontal fashion. If more than one box is used in the Front Panel, the boxes are stacked on top of each other.
- **Switch** - The switch, such as the one pictured in Figure 18, is a specialized container that can contain controls that are positioned around the workspace buttons in the center. The workspace buttons are special controls that, when selected, switch the user between their various workspaces. There can only be one switch container in the Front Panel.

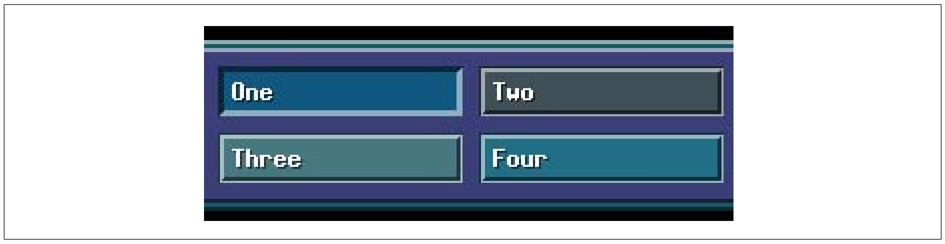


Figure 18. Front panel switch container

- **Subpanels** - A subpanel, such as the one shown in Figure 19 on page 67, is a slide-up container that can be attached to any of the controls in the Front Panel. If a Front Panel control has a subpanel attached to it, an upwards-pointing triangle will appear in the rectangle just above the control. When this rectangle is selected with a mouse click, the associated subpanel will slide up revealing its contents. The subpanel container is, generally, used to hold controls that you want to have easily available but do not have to be displayed all the time.

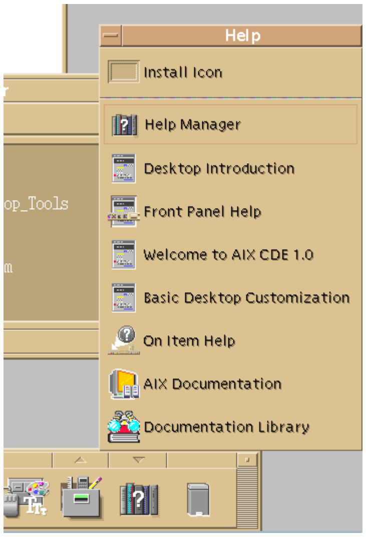


Figure 19. Default Front Panel Help subpanel container

5.1.3.2 Customizing an existing Front Panel

The Front Panel can be fully customized; you can:

- Add and remove controls in a Front Panel
- Add and remove subpanels in a Front Panel
- Add and remove controls in a subpanel
- Add, remove, and rename workspaces in a Front Panel
- Prevent users from changing controls and subpanels
- Create a new subpanel

There are two available methods for customizing an existing Front Panel:

- **Interactive** - This method uses the built-in customization facilities of the desktop to interactively make changes to the Front Panel.
- **Manual** - This method customizes the Front Panel by either:
  - Altering or overriding the Front Panel component definitions that are used to build the Front Panel
  - Changing X Window resources that specify front-panel parameters

5.1.4 Style Manager

The Style Manager component of the desktop allows the user to interactively customize the look and operation of their desktop environment. The Style Manager's main user interface is a graphical menu, pictured in Figure 20, and consists of icons that represent the available customization tools. These tools allow the user to customize the following characteristics:

- **Color** - Sets the color palette and the color usage for all desktop windows and supporting applications
- **Font** - Sets the font size used by text and labels in the desktop and supporting applications
- **Backdrop** - Sets the backdrops used by the available workspaces
- **Keyboard** - Sets keyboard characteristics for key click volume and key repeat rates
- **Mouse** - Sets the mouse characteristics for acceleration, double click, and handedness
- **Beep** - Sets the system speaker characteristics for beep volume, tone, and duration
- **Screen** - Sets the use and operation of a screen saver and screen lock
- **Window** - Sets window management characteristics for focus, window dragging visuals, and icon placement
- **Start-up** - Sets the characteristics of the Session Manager, for example, creating home sessions and which session to start by default
- **Workspaces** - Allows you to hide the workspace buttons in the Front Panel

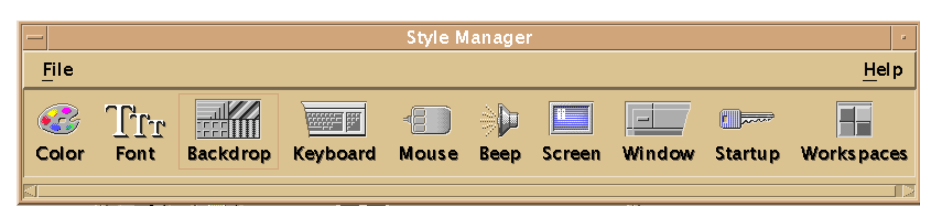


Figure 20. Style Manager main user interface

5.1.5 Workspace Manager

The Workspace Manager manages the different workspaces available within CDE. Every time users create, delete, or change windows within CDE, they are interacting with the Workspace Manager. It is the Workspace Manager that defines how windows look and feel and also how the desktop responds to user inputs, such as mouse and keyboard actions.

In CDE, it is possible to have many (four by default) workspaces. This allows the user to have separate applications running in separate workspaces without cluttering up the desktop. The Workspace Manager allows the user to easily switch between these workspaces.

For example, if a user wants to run three applications, such as Performance Toolbox, Framemaker, and Netscape, in the same workspace, it would soon become cluttered and hard to manage. With CDE, the user is able to put each application in a separate workspace and switch between them using either the Front Panel or the Graphical Workspace Manager (see Figure 21).

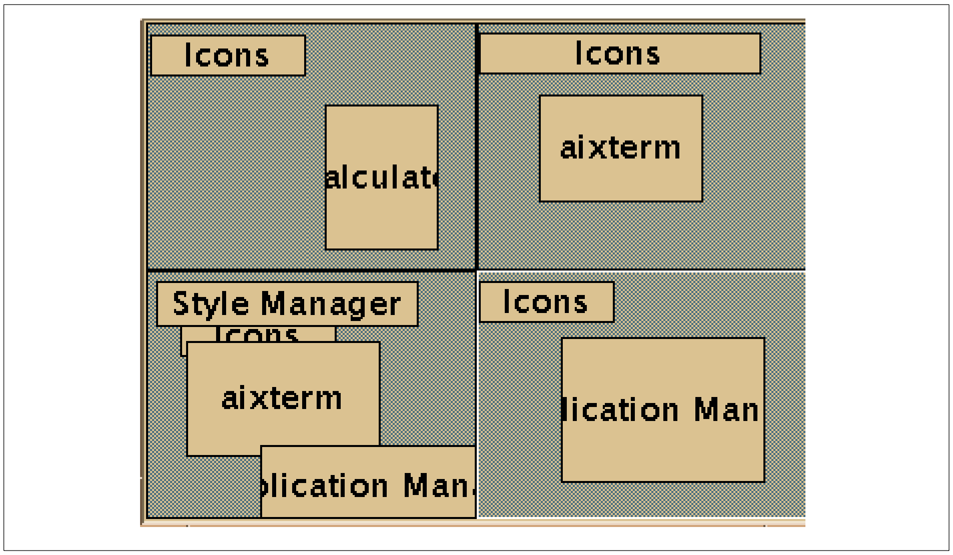


Figure 21. Graphical Workspace Manager

As with all CDE components, the features of the Workspace Manager can be customized by both the system administrator and individual users. This includes:

- The number of workspaces and their appearance
- The appearance of windows and icons in the workspace
- The actions associated with mouse buttons
- The actions associated with particular key stroke combinations

- Workspace and window pop-up or pull-down menu entries

The Graphical Workspace Manager can be called from the Front Panel or mouse menu. It shows a window with as many subwindows as you have workspaces on your screen. In each subwindow, you have rectangles associated with active application windows. These subwindows can be moved from one workspace to another.

Note

Only currently-running applications or programs can exist within a CDE workspace. You cannot, for example, have an icon to execute a program residing in a workspace; this icon would have to form part of the Front Panel. Applications can only be executed from the Front Panel or from other running applications. While this may be appear obvious, this is a fundamental difference between the CDE user interface and many other non-UNIX user interfaces.

5.1.6 File Manager

The CDE File Manager, shown in Figure 22 on page 71, is a tool that provides a graphical representation of the AIX file system. Since the various file types, such as directories, executables, graphical images, and so on, can have actions associated with them, they are known as objects. The File Manager can display a different icon for different types of files. For instance, a directory is represented by a folder, and a graphical image is represented by an icon of the Mona Lisa with a tag to indicate the graphic type (GIF, JPEG, and so on). This allows the user to browse the file system in a friendly and intuitive fashion.

Within the File Manager, certain file types can have applications associated with them. For example, a text file can be associated with an editor so that, when the user clicks on the text file, the File Manager will automatically open the editor with the selected file loaded.

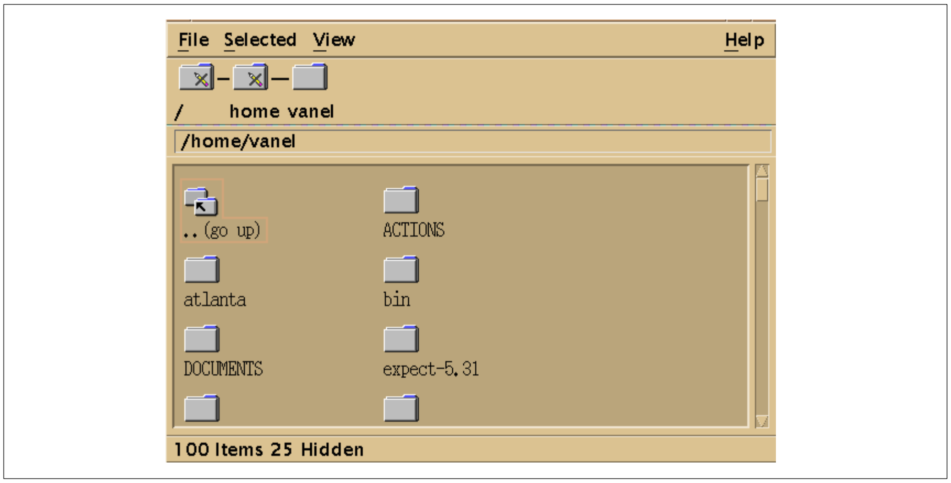


Figure 22. File Manager

5.1.6.1 Objects

Within CDE, an object is merely a file with several properties associated with it. For example, a directory will have the following properties attached:

- An icon - By default, the image of a folder.
- A name.
- A set of associated actions - For example, if the directory is clicked, it will be opened; if a file is dropped on it, the file will be copied into the directory.

These properties can either be system defaults or user specified. The system administrator can also create additional default object types as required.

5.1.7 Application Manager

The Application Manager component of CDE, shown in Figure 23 on page 72, is a graphical menu system similar in appearance to the File Manager. It allows users to execute their applications without the need to know where in the file system the application is located or even which machine the application is on. The user simply clicks on the required application, and it is loaded.

The Application Manager is actually a special instance of the File Manager with the restriction that you cannot move outside its directory. For example, changing the visual representation of objects or executing an object is the same in the Application Manager as it is in the File Manager.

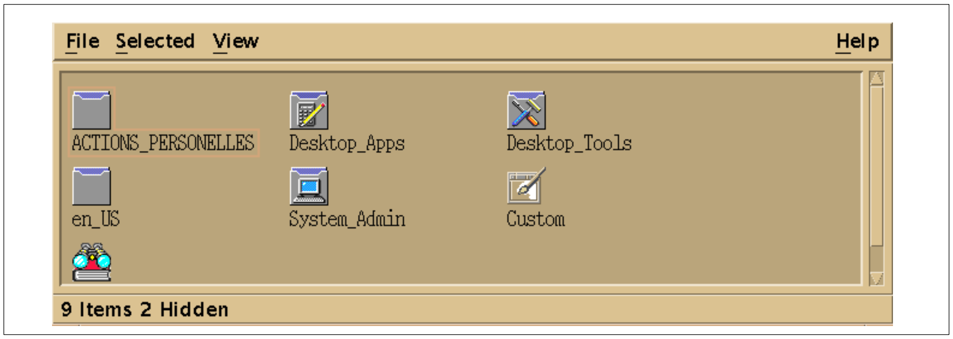


Figure 23. Application Manager

As mentioned earlier, the Application Manager provides a repository of various applications and tools. These applications and tools are arranged similarly to the File Manager. Applications and tools are stored in groups similar to directories, which, in turn, can hold other groups or the icons for the applications themselves. The main difference between the File Manager and the Application Manager is that applications and tools are organized by functionality and not by file system location. Applications and tools can easily be added to the Application Manager.

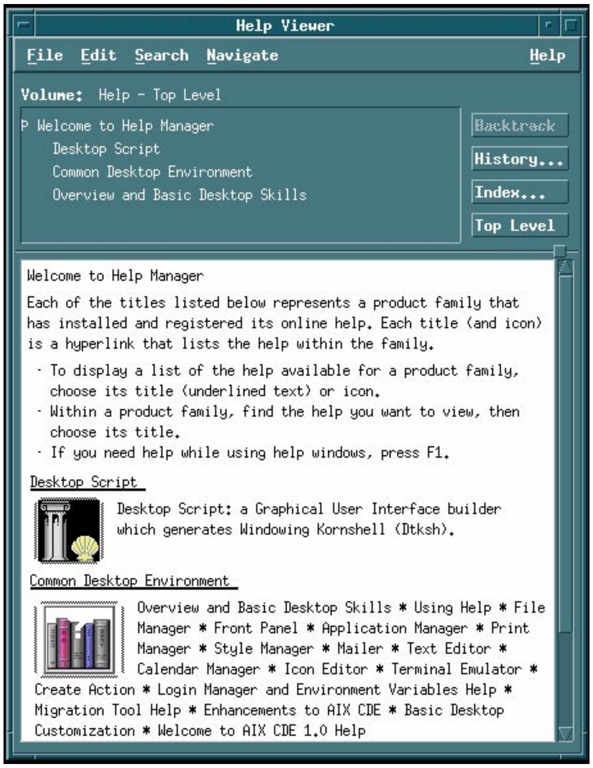
5.1.8 Help Manager

The Help Manager component of CDE, shown in Figure 24 on page 73, is the system through which the user can seek assistance for using the desktop or one of its components; information is in a hypertext format. This provides users with a friendly and intuitive way of either answering particular queries on CDE or teaching themselves more about CDE.

The information is organized hierarchically by:

1. **Volume** - A help volume consists of groups of related help topics, usually covering a particular product, tool, or task.
2. **Topic** - A help topic can be either a subgrouping of information pertinent to a particular topic in the help volume or the entry containing information on the topic itself. Help topics are arranged hierarchically. The first topic in a help volume is called the *home* topic.

The Help Manager allows you to navigate through the different topics and volumes via hyperlinks in the help information.



Volume: Help - Top Level

P Welcome to Help Manager
Desktop Script
Common Desktop Environment
Overview and Basic Desktop Skills

Welcome to Help Manager

Each of the titles listed below represents a product family that has installed and registered its online help. Each title (and icon) is a hyperlink that lists the help within the family.

- To display a list of the help available for a product family, choose its title (underlined text) or icon.
- Within a product family, find the help you want to view, then choose its title.
- If you need help while using help windows, press F1.

Desktop Script

Desktop Script: a Graphical User Interface builder which generates Windowing Kornshell (Dtksh).

Common Desktop Environment

Overview and Basic Desktop Skills * Using Help * File Manager * Front Panel * Application Manager * Print Manager * Style Manager * Mailer * Text Editor * Calendar Manager * Icon Editor * Terminal Emulator * Create Action * Login Manager and Environment Variables Help * Migration Tool Help * Enhancements to AIX CDE * Basic Desktop Customization * Welcome to AIX CDE 1.0 Help

Figure 24. Help Manager interface

5.2 Windows 2000 user interface

The Windows 2000 user interface is similar to the Windows 98 and Windows NT user interfaces. This section gives an overview of this user interface and highlights its main features.

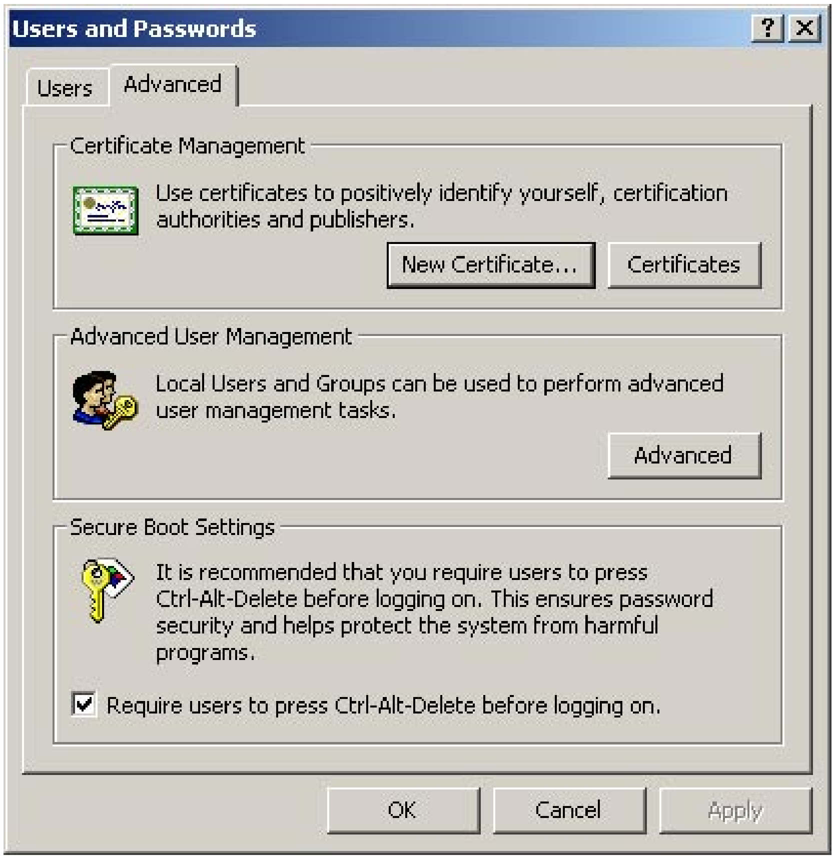
Windows 2000 uses the concept of objects in its desktop; every item on the desktop is an object. For example, a word processor document is an object, and, as such, it has a name, an associated icon, and a series of actions or reactions to user input. If you clicked on such a document, the associated word processor would be opened with the selected document loaded.

Because all items under Windows 2000 are objects, they all have properties that can be changed to affect the way they appear, where they appear, and how they respond to user input. To change an object’s properties, simply place the mouse cursor on the object and click the right mouse button (or press Alt+Enter). This will display the Properties menu for that particular object.

5.2.1 Logging on

To log on to a workstation or server running Windows 2000, you have to simultaneously press the Ctrl, Alt, and Delete keys. By pressing this key sequence, you are ensuring that the login prompt displayed is authentic and not a virus. For more details on this and other Windows 2000 security features, see Chapter 7, “Security” on page 153. Having pressed these keys, you will be shown the login screen. Once you have entered your login name and password, you will be presented with the Windows 2000 desktop.

For home or low security environments, the three finger salute (Ctrl, Alt, Delete) can be switched off in the **Control Panel** -> **Users & Passwords** applet, seen in Figure 25 on page 75. It is even possible to turn off the login procedure completely, sending who ever turns on the computer directly to the Windows 2000 desktop.



Certificate Management

Use certificates to positively identify yourself, certification authorities and publishers.

Advanced User Management

Local Users and Groups can be used to perform advanced user management tasks.

Secure Boot Settings

It is recommended that you require users to press Ctrl-Alt-Delete before logging on. This ensures password security and helps protect the system from harmful programs.

☑ Require users to press Ctrl-Alt-Delete before logging on.

Figure 25. Advanced Users and Passwords applet

5.2.2 Windows 2000 Active Desktop

Each user under Windows 2000 may tailor the Windows 2000 desktop to his or her own requirements including such changes as the color scheme, backdrop, sound schemes and so on.

Under Windows 2000, when a user changes a particular setting, that setting is saved in a profile for the user. When the user logs back in at a later date, his or her desktop will be restored using the user’s profile.

If the profile is put on a server instead of on the local hard drive, the profile will follow the user wherever he logs on. This is called roaming user profiles and can be a very powerful feature if implemented correctly.

Windows 2000 does not save the status of running applications when you log out; therefore, it cannot automatically start applications that you were previously running when you log back in. However, if you have unsaved data in an application, you will be prompted to save it.

Nevertheless, you can specify to automatically start certain applications. For example, if you always use a Web browser and a word processor, you could have them automatically start when you log in.

The Windows 2000 desktop consists of two main components: The taskbar and the desktop.

A new functionality in Windows 2000 enables you to add your favorite Internet Web pages directly to your desktop and browse through them in the same way as they would be in the Microsoft Internet Explorer, called Active Desktop. Actually, this was introduced with the release of Internet Explorer 4 but is now an integral part of Windows 2000. You can enable manual or automatically scheduled synchronization of these pages with their original location depending on whether you are working on- or off-line.

5.2.2.1 Taskbar

The Taskbar, shown in Figure 26, is a dynamic menu that contains an icon or button for every active application or open document in the system. This picture shows that the Adobe FrameMaker application is running in a window. You can switch between open applications or documents very easily by simply clicking on the appropriate icon on the Taskbar.

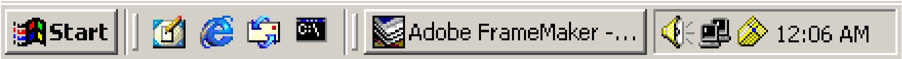


Figure 26. Windows 2000 Taskbar

The Taskbar areas are from left to right: Start button, Quick Launch bar, Taskbar, and the status area containing tray icons for utilities. In this case we have the volume control, LAN icon, Norton Antivirus and the clock display.

By default, the Taskbar is occupying the lower part of your screen but can easily be moved by dragging and dropping the taskbar itself on any of the other three sides of your screen.

There is also an option to hide the taskbar until you move the mouse over it.

5.2.2.2 Start button

As its name implies, the Start button is one of the fundamental controls of the Windows 2000 user interface. From here, the user can start applications available on the system, log off or shut down the system. By pressing the Start button, the user can intuitively navigate through the entire Windows 2000 system and select the application, document or resource of his or her choice.

When pressing the Start button, the first, or root menu, gives the user several options:

- **Windows Update** - This starts Microsoft Internet Explorer, and, if you are connected to the Internet, opens the following URL:

  `http://windowsupdate.microsoft.com/`

  From here, you can download the latest Windows updates.
- **Programs** - This takes the user through a set of program menus of installed applications allowing users to start an application of choice.
- **Documents** - Allows the user to select from a submenu containing links or shortcuts to the most recently used documents.
- **Settings** - This allows the user to change almost any aspect of the system, provided that they have sufficient privileges.
- **Search**- This allows the user to search for files, computers, printers or a generic Internet search using a search engine of choice.
- **Help** - This starts the help system.
- **Run** - This allows the user to run a command by entering its name.
- **Log Off Administrator** - After the confirmation question, this closes your desktop session and logs you off as the Administrator.
- **Shutdown** - This allows the user to (if authorized) shut down the system, restart the system, or log off.

The layout of the Start menu can vary drastically in design depending on user settings and active system and user policies.

5.2.2.3 The Quick Launch bar

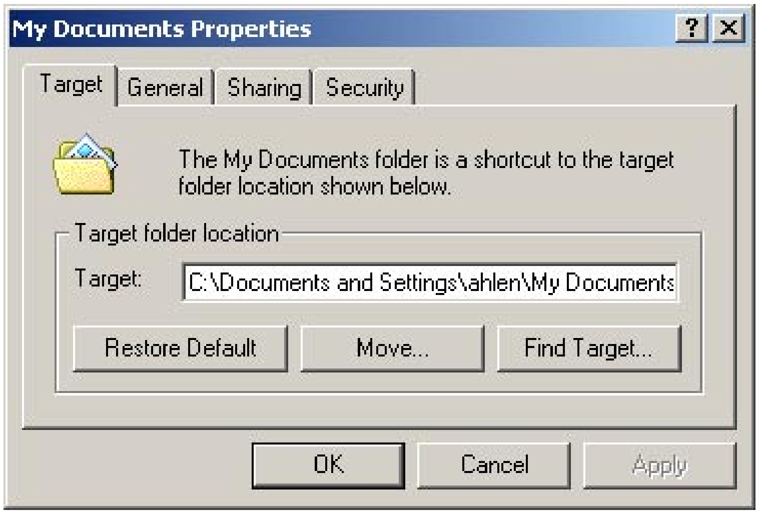
This is the area on the Taskbar just to the right of the Start button. You can start any application that has the icon in the Quick Launch bar by clicking it once with the left mouse button. You can add other application icons to the Quick Launch bar by simply dragging and dropping.

5.2.2.4 The Desktop

The Windows 2000 Desktop is basically everything on the screen except the Taskbar. The Desktop can contain applications or shortcuts to applications. For instance, if you have an application that you use regularly, instead of always going through the Start menu to access it, you could create a shortcut to the application on the desktop and use it to launch the application. Adding it to the Quick launch bar would accomplish the same thing.

By default, the Windows 2000 Desktop contains five objects:

- **My Documents** - This application opens a window from which you can access your documents. The My Documents folder is considered a ‘special’ folder, which means that it has properties that regular folders do not have as illustrated in Figure 27.



My Documents Properties

The My Documents folder is a shortcut to the target folder location shown below.

Target folder location

Target: C:\Documents and Settings\ahlen\My Documents

Figure 27. My Documents folder properties

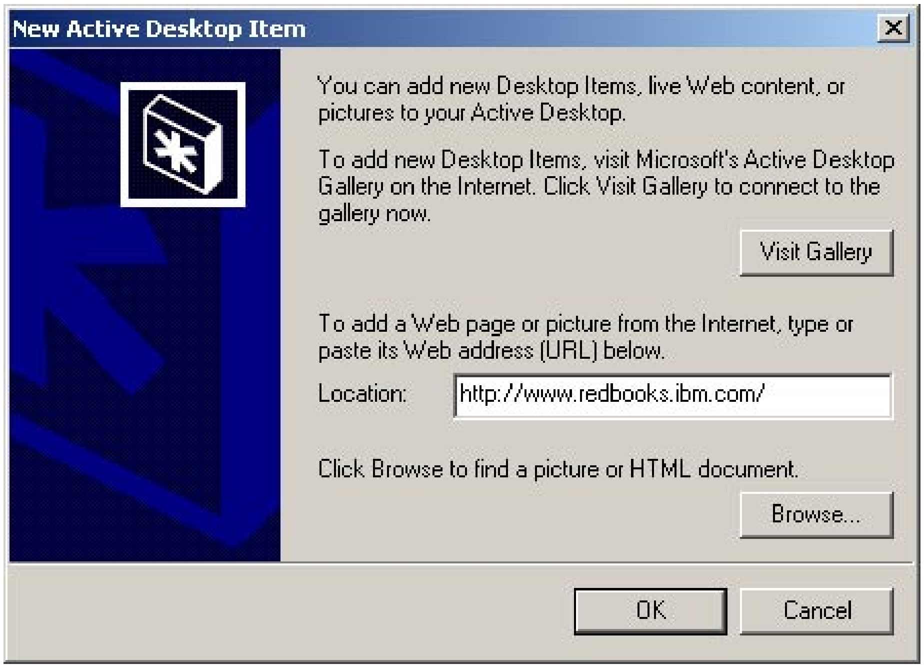
- **My Computer** - This application opens a window from which you can access your storage devices, such as local or remote disk drives, and access the control panel.
- **My Network Places** - This provides a window into your network environment. From here, you can browse your LAN or WAN and share files or printers with all compatible services on the network.
- **Recycle Bin** - The recycle bin contains deleted files or objects. This enables you to salvage items that may have been deleted by mistake.
- **Internet Explorer** - This starts the Microsoft Internet Explorer Web browser.

The desktop also contains active application windows, unless they have been minimized. Under Windows 2000, there is no facility to support multiple desktops, which can sometimes lead to an overcrowded desktop.

There is, however, support for multiple monitors, so if you have several graphics adapters or one multi monitor adapter in your machine, you could connect an additional monitor, using it as a parallel desktop.

5.2.2.5 Active Desktop

You can add your favorite Web pages to the active desktop by clicking the right mouse button on the empty space in the workplace and choosing the options **Active Desktop** -> **New Desktop Item**. As shown in Figure 28, a window will open, and you will add your favorite URL to the Location field.



New Active Desktop Item

You can add new Desktop Items, live Web content, or pictures to your Active Desktop.

To add new Desktop Items, visit Microsoft's Active Desktop Gallery on the Internet. Click Visit Gallery to connect to the gallery now.

To add a Web page or picture from the Internet, type or paste its Web address (URL) below.

Location: http://www.redbooks.ibm.com/

Click Browse to find a picture or HTML document.

Figure 28. Adding a new desktop item

After positioning the window on the screen, your desktop will look like the one displayed in Figure 29 on page 80. The displayed Web page is active and you can browse through it the same way as if it was displayed in Microsoft Internet Explorer.

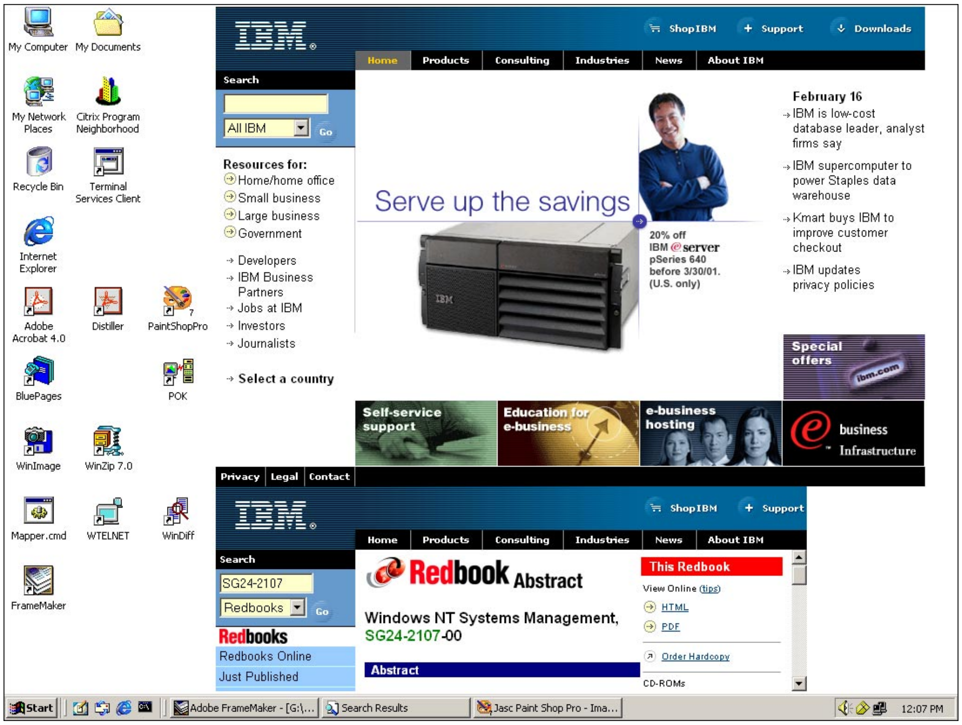
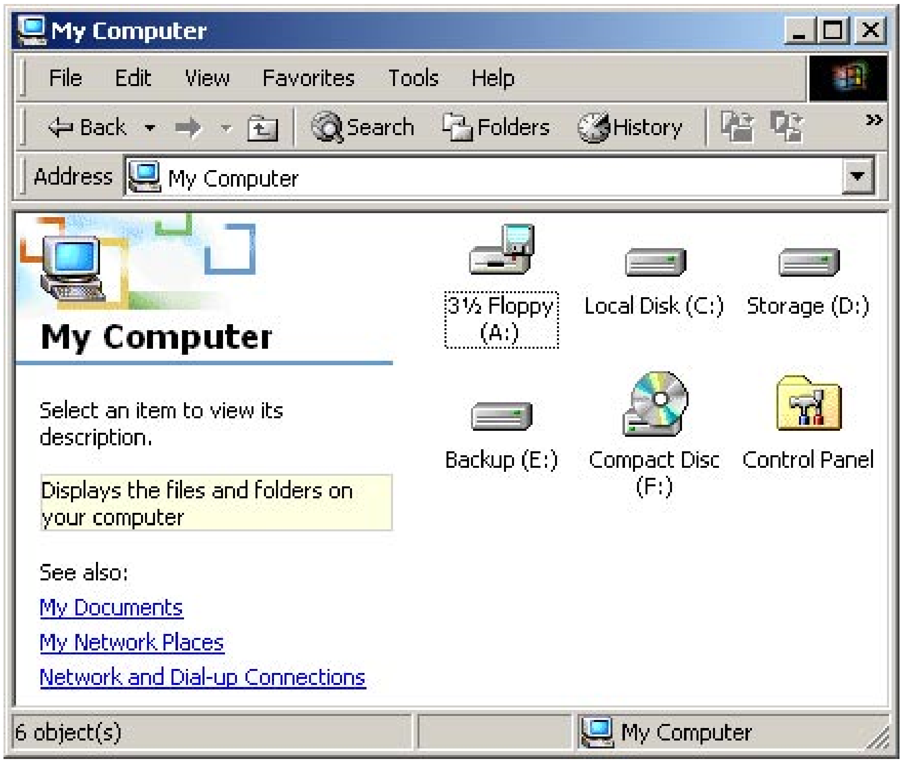


Figure 29. Active desktop

5.2.2.6 My Computer

By selecting the My Computer application from the desktop, you can manipulate all the objects in your system, like your disk drives and the Control Panel. My Computer offers you an intuitive method of exploring your system. When you start My Computer, you are shown an iconic representation of your system’s disk drives and a folder labeled Control Panel (see Figure 30 on page 81).



My Computer

Select an item to view its description.

Displays the files and folders on your computer

See also:

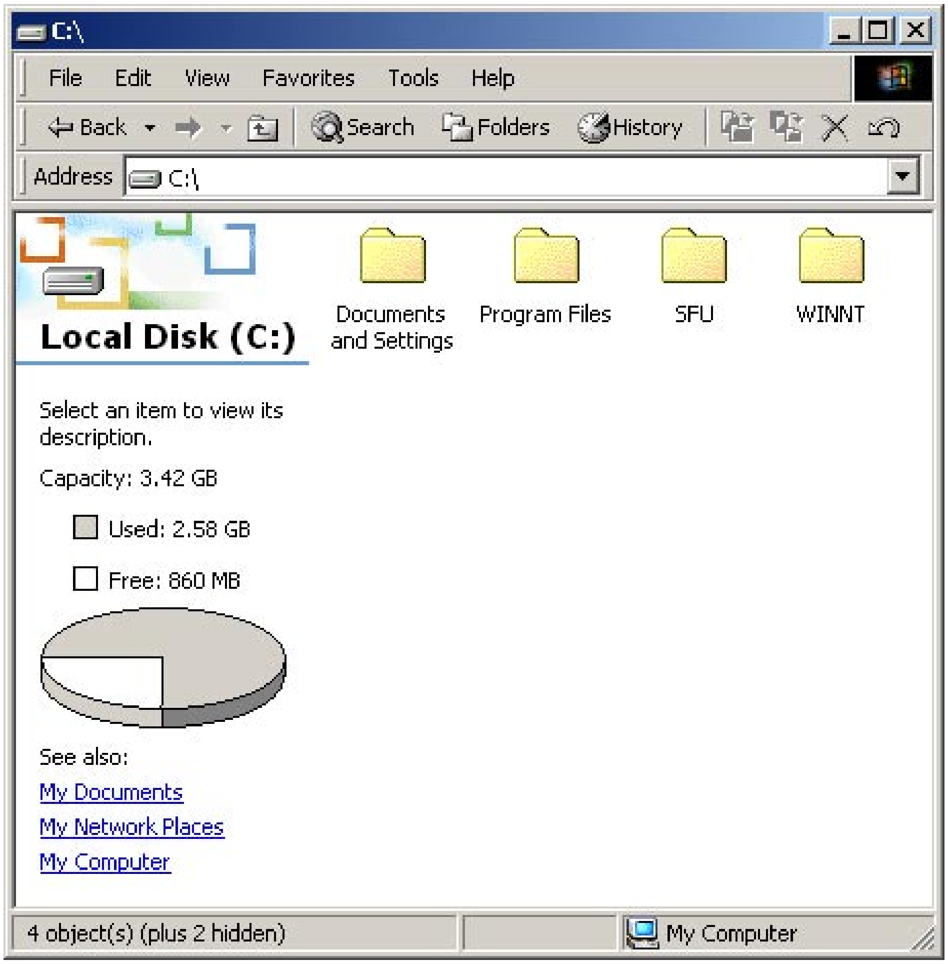
My Documents

My Network Places

Network and Dial-up Connections

Figure 30. My Computer

If you double click the Local Disk (C:) icon, the window will display the contents of the C drive’s root directory (see Figure 31 on page 82).



Select an item to view its description.

Capacity: 3.42 GB

□ Used: 2.58 GB

□ Free: 860 MB

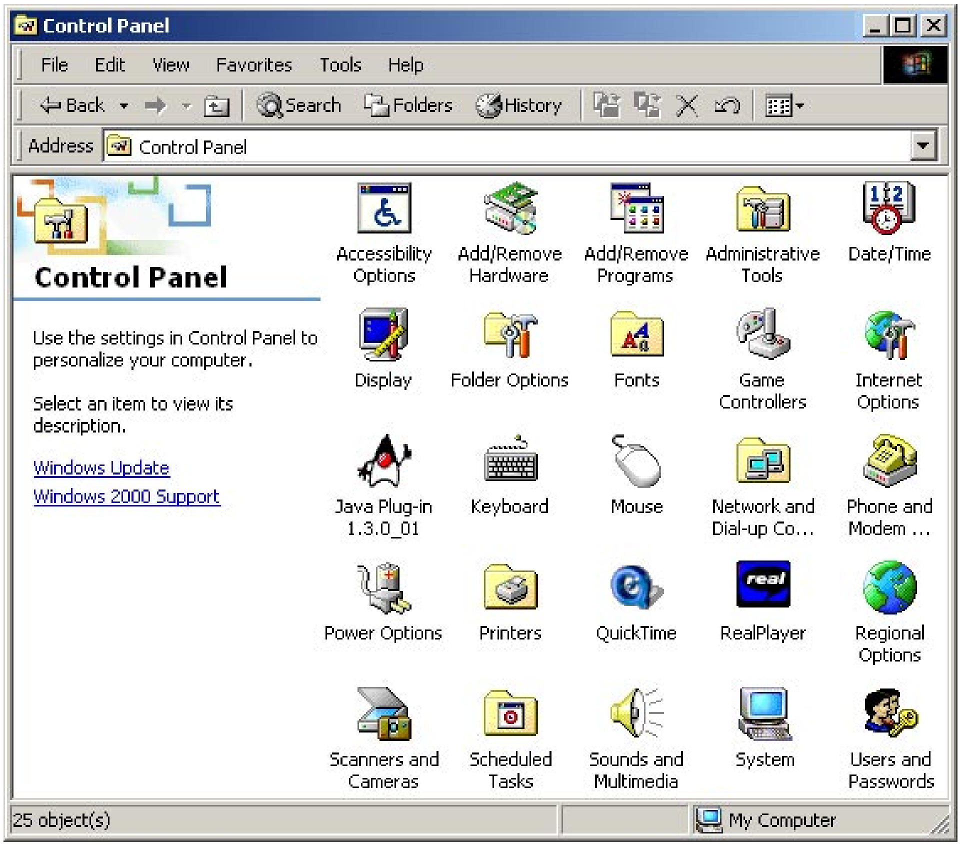
4 object(s) (plus 2 hidden)

Figure 31. Viewing a drive using My Computer

By selecting and clicking objects, the user can navigate through the Windows 2000 system. With the back and forward buttons the user can display previous windows in the same manner as in Microsoft Internet Explorer.

5.2.2.7 Control Panel

The Windows 2000 Control Panel allows you to configure the hardware and software on the system you are using. The Control Panel looks like any other folder and displays the different applets like applications as seen in Figure 32 on page 83.



Control Panel

Use the settings in Control Panel to personalize your computer.

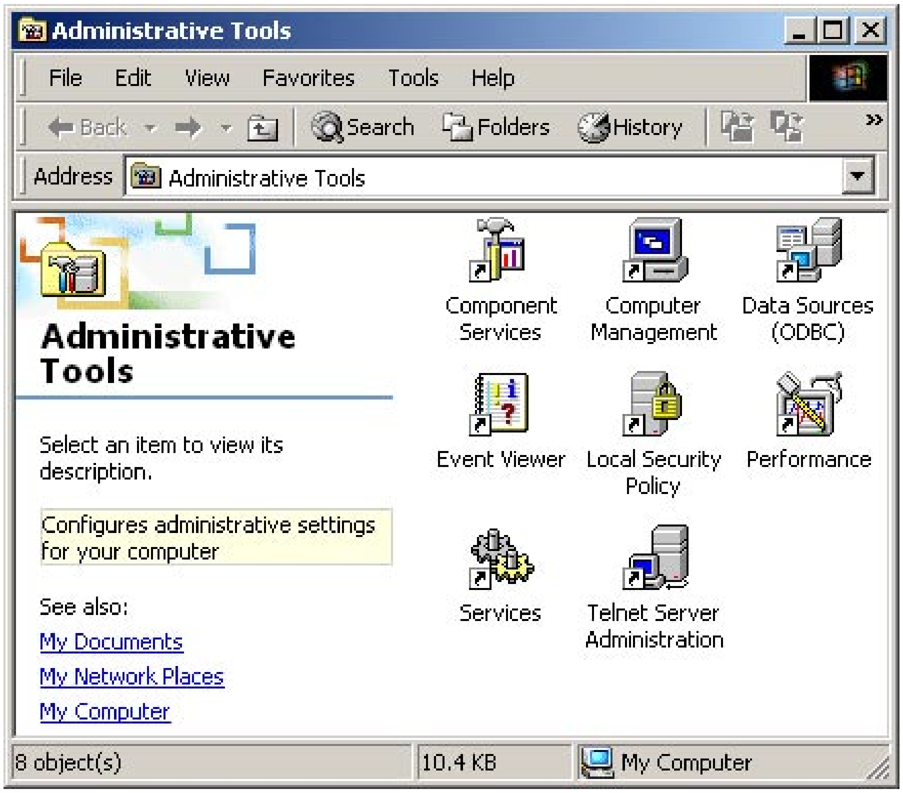
Select an item to view its description.

Windows Update

Windows 2000 Support

Figure 32. The Control Panel

Additional system management tools can be found by opening the Administrative Tools folder in the Control Panel window (see Figure 33 on page 84).



Administrative Tools

Select an item to view its description.

Configures administrative settings for your computer

See also:
My Documents
My Network Places
My Computer

Figure 33. Administrative tools

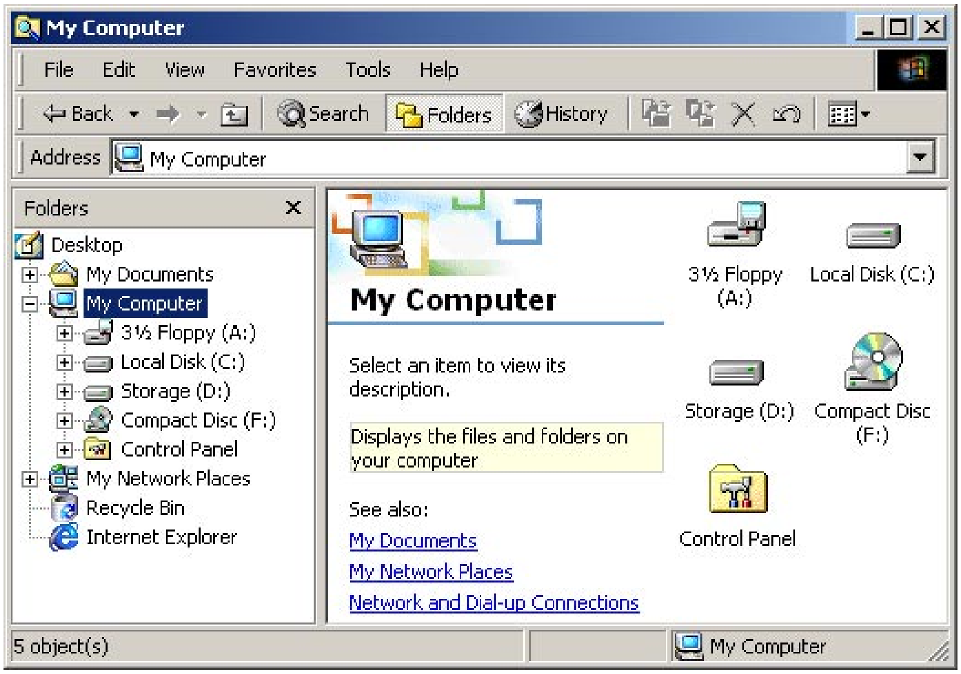
Note

For more information on the Windows 2000 system management interface, refer to section 8.2.4, “Windows 2000 System Administration Interface” on page 296.

5.2.2.8 The Windows 2000 Explorer

The Windows Explorer allows you to access the same programs, files, and resources as My Computer, but it does so using a different user interface. If you look at Figure 34 on page 85, you can see the folder hierarchy of your system in the left-hand window.

The Windows Explorer allows the user to map a network drive. This is equivalent of mounting a remote Logical Drive or shared directory and seeing it as a Local Logical Drive, allowing the user to manage local and remote files transparently.



Select an item to view its description.

Displays the files and folders on your computer

Figure 34. Windows Explorer

5.2.2.9 Windows 2000 Help

Windows 2000 contains a very powerful help system, shown in Figure 35 on page 86. Besides the standard help functions, such as the ability to search for a topic, a user can also be guided, by pointing and clicking, through most tasks on the system. Particularly useful for Windows NT users are the *New features* and *New ways to do familiar tasks* topics.

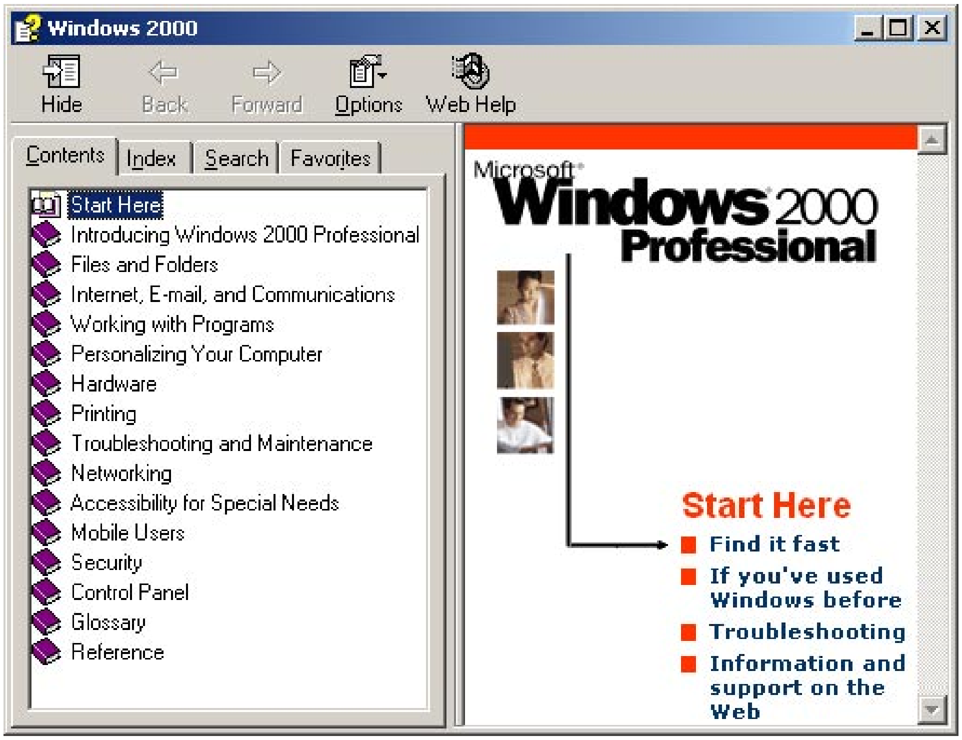

Windows 2000
Hide
Back
Forward
Options
Web Help
Contents
Index
Search
Favorites
Start Here
Introducing Windows 2000 Professional
Files and Folders
Internet, E-mail, and Communications
Working with Programs
Personalizing Your Computer
Hardware
Printing
Troubleshooting and Maintenance
Networking
Accessibility for Special Needs
Mobile Users
Security
Control Panel
Glossary
Reference
Microsoft Windows 2000 Professional
Start Here
Find it fast
If you've used Windows before
Troubleshooting
Information and support on the Web

Figure 35. Windows 2000 help system

5.3 Graphics Application Programming Interfaces (APIs)

This section examines the graphics APIs available to AIX and Windows 2000. There may be additional third-party APIs available, but these are beyond the scope of this section.

5.3.1 AIX

For customers requiring 3D capability, AIX provides the facilities for the development and execution of 3D applications using a variety of industry standard APIs. This includes hardware-accelerated support for graPHIGS and GL 3.2 as well as a pure software implementation of OpenGL. The following graphic APIs are available on AIX 5L:

OpenGL
OpenGL (Version 1.2) is a vendor-neutral application programming interface that provides advanced 2D and 3D graphics functions. It provides a hardware- and operating system-independent interface to graphics software.

OpenGL is typically used for engineering, visualization, simulation, and other graphics-intensive applications. OpenGL has become one of the two major graphic APIs available on AIX, along with graPHIGS. It is an easy to use, full-featured, and network transparent. OpenGL is available on AIX 5L base installation CDs. The AIX OpenGL package consists of a rendering library, utility toolkit, and network protocol.

GL 3.2

The Graphics Library (GL) API is a 3D API which was developed by Silicon Graphics, Incorporated as a proprietary API for use on their graphics platforms and this API has been licensed to IBM. It was once very popular and used by many independent software vendors and was the basis for their 3D applications. The strength of GL is immediate mode graphics and the biggest disadvantage of the GL API is that it does not provide interoperability in heterogeneous network environments and it is not easily portable even between the few platforms that support it. So, this API has almost become obsolete as a 3D graphics API since OpenGL has taken its place in the industry. AIX 5L still support this API for all previous 3D graphics adapters.

graPHIGS

graPHIS API is IBM's implementation of the ISO/ANSI Programmer's Hierarchical Interactive Graphics System (PHIGS) standard, which includes a subset of the PHIGS PLUS extensions (such as independent lighting and shading controls, advanced primitives, and extended rendering attributes) as well as IBM's own extensions. The graPHIGS API is supported on IBM Enterprise Systems and the RS/6000 systems equipped with 2D or 3D adapters.

Softgraphics

Softgraphics allow all 3D functions to be performed by software where the graphics adapter is used simply as a frame buffer to display the image. This implementation makes it possible to run 3D applications on any 2D graphics adapter. Softgraphics performance scales with the performance of the RS/6000 processor. Softgraphics provides a uniform development environment for 3D applications on RS/6000s with entry-level graphics adapters.

5.3.2 Windows 2000

OpenGL

Windows 2000 provides native support for OpenGL as part of the base operating system. However, the preferred graphics API for Windows 2000 is the Microsoft proprietary DirectX standard.

DirectX
Microsoft DirectX is a suite of multimedia APIs that comes standard with Windows 2000. DirectX provides a standardized development platform for Windows-based PCs by enabling access to specialized hardware such as 3-D graphics acceleration chips and sound cards without having to write hardware-specific code. These APIs control 2-D graphics acceleration; support for input devices such as joysticks, keyboards, and mice; and control of sound mixing and sound output.

The following components make up the DirectX APIs:

- DirectDraw
- Direct3D
- DirectInput
- DirectSound
- DirectPlay
- DirectShow
- DirectMusic

The components that make up DirectX provide hardware manufacturers with a flexible platform much like OpenGL, except that DirectX covers more than the graphics subsystem.

Windows 2000 comes with Version 7 of DirectX. At the time of this writing, the latest version is DirectX 8.0a, which can be downloaded from Microsoft at the following URL:

```
http://www.microsoft.com/directx/homeuser/downloads/default.asp
```

5.4 Command line Interface

This section describes the AIX and Windows 2000 command line interfaces.

5.4.1 AIX shells

One of the main features of any UNIX operating system (and, as such, also of AIX) is the shell. It is both a command interpreter and a command programming language. The following section describes and highlights the main characteristics of AIX shells.

5.4.1.1 Shell overview

The shell is the outermost layer of the operating system. Shells incorporate a command programming language to control processes and files and to start and control other programs. The shell manages the interaction between you and the operating system by prompting you for input, interpreting that input for the operating system, and then handling any resulting output from the operating system.

Shells provide a way for you to communicate with the operating system. This communication is carried out either interactively (input from the keyboard is acted upon immediately) or as a shell script. A shell script is a sequence of shell and operating system commands that is stored in a file.

When you log in to the system, the system locates the name of a shell program to execute as specified by the system administrator. Once executed, the shell displays a command prompt. This prompt is, by default, a $ (dollar sign), but it is usually changed by the system administrator to something that is less financially oriented. When you type a command at the prompt and press the Enter key, the shell evaluates the command and attempts to execute it. Depending on your command instructions, the shell writes the command output to the screen or redirects it elsewhere. It then returns to the command prompt and waits for you to type another command.

A command line is the line on which you type. It contains the shell prompt. The basic format for each line is:

```
$command [argument]
```

Please note that UNIX operating systems are *case sensitive*.

The shell considers the first word of a command line (up to the first blank space) as the command and all words after that as arguments. This section discusses:

- Shell features
- Available shells
- Shells terms
- Creating and running a shell script
- Korn shell or POSIX shell
- Bourne shell
- C shell

5.4.1.2 Shell features

The primary advantages of interfacing the system through a shell are:

Wildcard substitution in file names (pattern matching)

This carries out commands on a group of files by specifying a pattern to match rather than an actual file name.

Background processing

This sets up lengthy tasks to run in the background, freeing the terminal for concurrent interactive processing.

Command aliasing

This gives an alias name to a command or phrase. When the shell encounters an alias on the command line or in a shell script, it substitutes the text to which the alias refers.

Command history

This records the commands you enter in a history file. You can use this file to easily access, modify, and reissue any listed command.

Command line editing

The shell allows you to specify your favorite editor to be used when editing the command line. This allows you to recall any command from the command history (see ***Command history***) and edit or change it. By using an advanced editor, such as vi or emacs, you can also search for previous commands.

File name substitution

This automatically produces a list of file names on a command line using pattern-matching characters.

Input and output redirection

This redirects input away from the keyboard and redirects output to a file or device other than the terminal. For example, input to a program can be provided from a file and redirected to another file or to a device.

Piping

This links any number of commands together to form a complex program. The standard output of one program becomes the standard input of the next.

Shell variable substitution

This stores data in user-defined variables and predefined shell variables.

Functions

This increases programmability and allows storing shell code in memory.

Integrated programming features

Several AIX programs are available for text manipulations, and some of this functionality has been included in the shell itself. Debugging primitives have also been added.

Advanced I/O features

This includes the ability to have two-way communication with a concurrent process. For further details, see section 5.4.1.7, “Process handling” on page 97.

Note

Not all of the features are available in all different shells. For example, in the Bourne shell, there is no Command History.

The following shells are provided with AIX 5L:

- Korn shell (started with the `ksh` command)
- Bourne shell (started with the `bsh` command)
- Restricted shell (a restricted version of the Bourne shell started with the `Rsh` command)
- C shell (started with the `csh` command)
- Trusted shell (a limited version of the Korn shell started with the `tsh` command
- Remote shell (started with the `rsh` command)

The login shell refers to the shell loaded when you log in to the computer system and is specified by your system administrator. The Korn shell is the standard AIX system login shell and is backwardly compatible with the Bourne shell.

The default or standard shell refers to the shell linked to (and started with) the `/usr/bin/sh` command. The Korn shell, also known as the POSIX shell, is set up as the default shell. The POSIX shell is called by the `/usr/bin/psh` command and is actually a link to the `/usr/bin/ksh` command.

5.4.1.3 Shell terms

The following definitions are helpful in understanding shells:

Built-in command

This is a command that the shell executes without searching for it and creating a new process.

Command
This is a sequence of characters in the syntax of the shell language. The shell reads each command and carries out the desired action either directly or by invoking separate utilities.

List
This is a sequence of one or more pipelines separated by one of these four symbols: ; (semicolon), & (ampersand), && (double ampersand), or || (double bar). Optionally, the list is ended by one of the following symbols: ; (semicolon), & (ampersand), or |& (bar and ampersand).

&
This asynchronously processes the preceding pipeline. The shell carries out each command in turn, processing the pipeline in the background without waiting for it to complete.

|&
This asynchronously processes the preceding pipeline and establishes a two-way pipe to the parent shell. The shell carries out each command in turn, processing the pipeline in the background without waiting for it to complete. The parent shell can read from and write to the standard input and output of the spawned command by using the `read -p` and `print -p` commands. Only one such command can be active at any given time.

> **Note**
>
> The `|&` symbol is valid only in the Korn shell.

&&
This processes the list that follows this symbol only if the preceding pipeline returns an exit value of 0 (zero), that is, if it completes successfully.

||
This processes the list that follows this symbol only if the preceding pipeline returns a nonzero exit value, that is, if the command fails.

Metacharacter
Each metacharacter has a special meaning for the shell and causes termination of a word, unless it is quoted. Metacharacters are: | (pipe), & (ampersand), ; (semicolon), < (less-than sign), > (greater-than sign), ((left parenthesis),) (right parenthesis), $ (dollar sign),‘ (backquote), / (backslash), ′ (right quote), ″ (double quotation marks), new-line character, space character, and tab character. All characters enclosed between single quotation marks are considered quoted and are interpreted literally by the

shell. The special meaning of metacharacters is retained only if they are not quoted. (Metacharacters are also known as parser metacharacters in the C shell.)

Pipeline
This is a sequence of one or more commands separated by a | (pipe). Each command in the pipeline, except possibly the last command, is run as a separate process. However, the standard output of each command that is connected by a pipe becomes the standard input of the next command in the sequence. If a list is enclosed with parentheses, it is carried out as a simple command that operates in a separate subshell. If the reserved word ! (exclamation point) does not precede the pipeline, the exit status will be the exit status of the last command specified in the pipeline. Otherwise, the exit status is the logical NOT of the exit status of the last command. In other words, if the last command returns zero, the exit status will be 1. If the last command returns greater than zero, the exit status will be zero.

Redirection
This redirects input away from the keyboard and redirects output to a file or device other than the terminal. For example, input to a program can be provided from a file and redirected to the printer or to another file.

Shell variable
This is a name or parameter to which a value is assigned. You can assign a variable by typing the variable name, an = (equal sign), and then the value (`MY_VAR=5`). The variable name can be substituted for the assigned value by preceding the variable name with a $ (dollar sign). Variables are particularly useful for creating a short notation for a long path name, such as HOME for the home directory. A predefined variable is one whose value is assigned by the shell. A user-defined variable is one whose value is assigned by a user.

Simple command
This is a sequence of optional parameter assignment lists and redirections, in any sequence. They are optionally followed by commands, words, and redirections. They are terminated by`;`, `|`, `&`, `||`, `&&`, `|&`, or a by newline character. The command name is passed as parameter `0` (as defined by the exec subroutine). The value of a simple command is its exit status, that is, zero if it exits normally or nonzero if it exits abnormally.

Subshell
This is a shell that is running as a child of the login shell or the current shell.

Wildcard characters

These are also known as pattern-matching characters. The shell associates them with assigned values. The basic wildcards are ?, *, [set], and [!set]. Wildcard characters are particularly useful when performing file name substitution.

5.4.1.4 Shell scripts

Shell scripts provide an easy way to carry out tedious commands, large or complicated sequences of commands, and routine tasks. A shell script is a file that contains one or more commands. When you type the name of a shell script file, the system executes the command sequence contained by the file.

You can create a shell script using a text editor. Your script can contain both operating system commands and built-in shell commands.

5.4.1.5 Commands and Built-in Commands

> **Note**
>
> In the following sections, *the shell* refers to the Korn shell.

Command history

The shell saves commands entered from your terminal device to a history file. The shell accesses the commands of all interactive shells by using the same-named history file with the appropriate permissions. By default, the Korn shell or POSIX shell saves the text of the last 128 commands entered from a terminal device. Use the `fc` built-in command to list or edit portions of the history file. To select a portion of the file to edit or list, specify the number or the first character or characters of the command. You can specify a single command or a range of commands.

Commands

A shell command is one of the following:

- Simple or built-in command
- Pipeline
- List
- Compound command
- Function

When you issue a command in the Korn shell or POSIX shell, the shell evaluates the command and acts in the following way:

1. It makes all indicated substitutions and determines whether the command is a special built-in command, and, if it is, the shell runs the command within the current shell process.
2. It compares the command to user-defined functions. If the command matches a user-defined function, the positional parameters are saved and then reset to the arguments of the function call. When the function completes or issues a return, the positional parameter list is restored, and any trap set on EXIT within the function is carried out. The value of a function is the value of the last command executed. A function is carried out in the current shell process. If the command name matches the name of a regular built-in command, that regular built-in command will be invoked.
3. It creates a process and attempts to carry out the command by using the `exec` command (if the command is neither a built-in command nor a user-defined function).

Korn shell compound commands

A compound command can begin with a reserved word, or a list of simple commands, or a pipeline. Usually, you will use compound commands, such as if, while, and for, when you are writing shell scripts.

Functions

The reserved word function defines shell functions. The shell reads and stores functions internally. Alias names are resolved when the function is read. The shell executes functions in the same manner as commands, with the arguments passed as positional parameters. The Korn shell or POSIX shell executes functions in the environment from which functions are invoked by sharing variable values and attributes, working directories, aliases, function definitions, special parameters, and open files. The exit status of a function definition is zero if the function was not successfully declared. Otherwise, it will be greater than zero. The exit status of a function invocation is the exit status of the last command executed by the function.

List of available shell commands

Table 2 contains a list of available shell commands and their actions.

Table 2. Built-in shell commands

Command	Action
`eval, exec, command, times`	Script execution and monitoring
`break, exit, continue, return, trap, wait, test`	Flow control
`set, typeset, unset, export, shift, readonly, let, getopts`	Variables and parameter manipulation
`newgroup, umask, ulimit`	User attributes and security
`print, read, echo`	I/O operations
`fg, bg, kill, jobs`	Job management
`alias, unalias, pwd, cd, fc`	Utility

5.4.1.6 The shell as a system programming environment

The shell has as many advanced programming capabilities as any command interpreter and can be used as a complete software environment. It is specially suited for writing programs to manage or automate system administrator tasks. A script or file that contains shell commands is a shell program. The shell has all the pieces you need:

Variables and arrays

As in any programming language, you can define variables or arrays, but the shell places heavy emphasis on character strings.

String operators

These allow you to manipulate values of variables without having to write full-blown programs. For example,

```
${ varname: +letter}
```

If varname exists and it is not null, return its value; otherwise, set it to letter and then return its value.

Pattern and regular expressions

Patterns are strings of characters that contain wildcard characters. The shell adds a set of operators, called regular-expressions, to manage them.

Flow controls

Flow controls give a programmer the power to specify that only one part of a program should run based on conditional values or that it should run repeatedly. The shell supports the following constructs, similar to a complete programming language:

- if/else
- for
- while
- until
- case
- select
- exit/return
- conditional expressions

I/O

The shell supports 16 I/O redirections, from `>` (redirect standard output), `<` (redirect standard input), and `> >` (append) to `|&` (background process with I/O from parent shell) and several I/O-level functions, such as read or print.

Debugging

Debugging programs is often closer to an art than an engineering process, and the AIX shell gives you some of the instruments you need to perform debugging operations, such as trace support (`xtrace`), fake signals (to simulate signals), or some debugging option with the `set` command.

5.4.1.7 Process handling

When a process is created, AIX assigns it a special number called a process ID (PID). The shell commands use the process ID to control processes. A process can be moved onto the background or foreground, and it can be suspended or monitored. You can also send signals or kill a process. A very important shell feature is the built-in `trap` command. A process within a shell script can be tailored to react to specific signals and to process them in their own way. A trap is like an interrupt routine that will be called by AIX when a specific signal has been sent to a process. For example:

```
trap ‘ echo “A Termination signal has been received”’ INT
while true
do
 echo “I am running .....”
 sleep 3
done
```

This shell script prints the phrase *I am running...* every three seconds, but it reacts with the message *A termination signal...* if someone tries to kill it. There is also a way to make sure that a script does not finish before a background command completes; this can be assured by using the `wait` built-in command. For example:

```
Program1 &
Program2
wait
```

where, if Program2 finishes first, the script will wait until Program1 ends. The execution of processes in parallel becomes very important when more than one processor is available and running time is a concern. In the previous example, the script's running time is, essentially, equal to that of the longest-running job, plus some overhead. The AIX shell provides several features to help programmers handle these issues. Two-way pipes, `|&` (command), and subshells are examples. For more information, see section 5.4.1.3, "Shell terms" on page 91.

5.4.1.8 Shell administration

The shell has some unique features for system customization other than the usual /etc/profile.

umask

This command lets you specify the default permissions that files have when users create them, and it contains the permissions that are turned off by default whenever a process creates a file.

ulimit

With this command, you can have fine grain control to avoid memory thrashing or infinite loop programs that result in poor utilization of system resources.

Table 3. Ulimit options

Parameter	Resource
`-a`	Lists all of the current resource limits.
`-c`	Specifies the size of core dumps in number of 512-byte blocks.
`-d`	Specifies the size of the data area in number of kilobytes.
`-f`	Sets the file size limit in blocks when the Limit parameter is used, or reports the file size limit if no parameter is specified. This is the default flag.

Parameter	Resource
-H	Specifies that the hard limit for the given resource is set. If you have root user authority, you can increase the hard limit. Anyone can decrease it.
-m	Specifies the size of physical memory in kilobytes.
-s	Specifies the stack size in kilobytes.
-S	Specifies that the soft limit for the given resource is set. A soft limit can be increased up to the value of the hard limit. If neither the -H nor -s flags are specified, the limit applies to both.
-t	Specifies the number of seconds to be used by each process.

Security

Security is always a problem, and the AIX shell provides a couple of features that help solve it:

- **Restricted shell** - The restricted shell is used to set up login names and execution environments whose capabilities are more controlled than those of the regular shell. The `Rsh` or `bsh -r` command opens the restricted shell. The behavior of these commands are identical to those of the `bsh` command, except that the following actions are not allowed:
  - Changing the directory (with the `cd` command)
  - Setting the value of PATH or other variables
  - Specifying path or command names containing a / (slash)
  - Redirecting output

  If the restricted shell determines that a command to be run is a shell procedure, it uses the Bourne shell to run the command. In this way, it is possible to provide an end-user with shell procedures that access the full power of the Bourne shell while imposing a limited menu of commands. This scheme assumes that the end-user does not have write and execute permissions in the same directory.
- **Tracked aliases** - Frequently, aliases are used as shorthand for full path names. One aliasing facility option allows you to automatically set the value of an alias to the full path name of a corresponding command. This special type of alias is a tracked alias. Tracked aliases speed execution by eliminating the need for the shell to search the PATH variable for a full path name. The `set -h` command turns on command tracking so that, each time

a command is referenced, the shell defines the value of a tracked alias. Since aliases take priority over executable files, the alias will always be run before any Trojan horse viruses that may exist on the system.

5.4.2 Windows 2000 command line interface

In this section, the Windows 2000 command line interface is referred to as the Windows 2000 command prompt.

5.4.2.1 Overview

The Windows 2000 command prompt is a character-based interface in which you enter commands from the keyboard. The command prompt is able to discriminate if you enter a DOS, OS/2, Win16, or POSIX character-based command. It spawns an instance of the appropriate environment subsystem and launches the application.

The Windows 2000 command prompt is mainly a superset of the DOS command prompt, but not all DOS commands have been ported to Windows 2000. For example, `ASSIGN` is no longer available. An alternative would be to use the `SUBST` command. Some other commands, such as `MIRROR` or `FASTOPEN`, have been removed because their functionality is available through other Windows 2000 services. Some UNIX-like functionality, such as && and ||, has been added to make UNIX users feel more at home and to allow for more complex scripts.

The Windows 2000 command prompt also allows you to scroll through a list of your previous commands and edit them using the standard DOS editing keys, such as Insert, Delete, and the arrow keys. Previously, this required the use of the `DOSKEY` command.

Note

The Windows 2000 command prompt has been implemented as an executable program called `CMD.EXE`.

The `COMMAND.COM` program still exists but using this will open a 16-bit subsystem and you might run into some problems running win32 applications from it.

An example of this is the output of the `SET` command. When executing the `SET` command from a `CMD.EXE` prompt and a `COMMAND.COM` prompt respectively, you will notice that `CMD.EXE` exposes more environment variables than `COMMAND.COM`. On the other hand, `COMMAND.COM` is the more ‘MS-DOS compatible’ version of the two shells. For example, this shell will call AUTOEXEC.NT and CONFIG.NT (located in the %SystemRoot%\System32 directory) just like `command.com` in MS-DOS used to call AUTOEXEC.BAT and CONFIG.SYS.

5.4.2.2 Commands

The Windows 2000 command prompt supports several categories of commands:

- Native
- Subsystem
- Configuration
- Network
- Utility

Note

It is possible to display the correct syntax for any command using the `HELP` command. For example, `HELP DIR` displays all the parameters for the `DIR` command. Typing `DIR /?` produces the same result.

Running the `HELP` command without any arguments lists all built-in commands along with a short description.

Native

These commands are built around the Windows 2000 command interpreter; some of them run external programs and others are built-in commands. For example, `DIR` is a built-in command; so, the code that will be executed is part of the Windows 2000 command interpreter. Other commands, such as `FORMAT`, are external programs; so, a FORMAT.COM executable is available in the system directory. These commands allow the user to perform file system operations, such as COPY, MOVE, ERASE, FORMAT, DIR and REN, start a new instance of the command interpreter, or simply search for a string by using the FIND command.

Configuration
These commands are used to customize the DOS environment and are useful when Windows 2000 runs the DOS subsystem.

Subsystem
These old DOS commands are no longer needed in Windows 2000, however they were retained for backward compatibility with existing DOS applications. For example, EDLIN, EXE2BIN, GRAPHICS, MEM, SETVER, and SHARE are subsystem commands.

Network
There are several network-related commands that can be executed from the command line and most of them begin with NET command plus a specific argument. The majority performs network management functions, such as NET SESSION, NET SHARE, or NET GROUP, and others allow you to send messages (NET SEND) or issue print jobs (NET PRINT).

5.4.2.3 AIX equivalents

Table 4 offers a basic comparison between some Windows 2000 commands and similar AIX commands. Note that Windows 2000 is not case sensitive and all commands have been listed in upper case.

Table 4. AIX/Windows 2000 commands

AIX	Windows 2000
at	AT
alias	DOSKEY
cat	TYPE
cd	CD
chmod	ATTRIB
cp	COPY
date	DATE
grep	FIND
ls	DIR

AIX	Windows 2000
echo	ECHO
man	HELP
mkdir	MKDIR
more	MORE
rm	DEL
set	SET
& (after the command)	START (before the command)

5.4.2.4 I/O redirection

Some I/O redirection capabilities are available within the command prompt interface, such as the possibility to redirect the input/output of a command to a file or a device by using the characters >, <, and >>. For example:

```
TYPE TEMP.TXT > LPT1 2>ERROR.TXT
```

will transfer the file, TEMP.TXT, to the line printer LPT1 and any error messages to the file error.txt. The pipe (“|”) is also supported; so, the output of the first command can be used as the input of the trailing command.

5.4.2.5 Batch files

Windows 2000 supports the concept of batch files (in UNIX jargon, this is known as shell scripts) to gather several commands into one and create simple automated procedures. Batch files must have the .BAT or .CMD extensions to be executed, and they can contain any Windows 2000 command or program file. For example:

```
COPY *.C %TEMP% /* copy all files with a .C extension in the */
                /* current directory to the temp directory */
DEL *.EXE /* erase all files with an .EXE extension from */
          /* the current directory */
```

A batch file accepts up to nine parameters. These parameters can be resolved with the %<number> keyword. For example, a file called DELFILE.BAT can be created:

```
DEL %1.OBJ
DEL %1.EXE
DEL %1.LNK
```

and can be executed by typing: DELFILE PROJECT, effectively deleting the PROJECT.OBJ, PROJECT.EXE and PROJECT.LNK from the current directory, if they exist. Inside a batch file, it is possible to read or write a system variable, such as PATH or create a local one. For example:

```
SET PATH=D:\WINNT\system32 /* set PATH value */
ECHO %PATH% /* print PATH value */
SET PATH=%PATH%; D:\temp /* add D:\temp directory to */
                        /* existing PATH value */
```

These variables only exist in the sessions where they are created, and system variable values affect applications running within the same session. If the default value of a system variable has to be changed, the System Control Panel Menu must be selected.

Batch files also allow you to control the execution flow with specific commands. These commands are listed in Table 5.

Table 5. Batch commands

Command	Action
CALL	Calls a batch file like a subroutine
FOR	Repeats a command
GOTO	Jumps to a label in the same file
IF	Executes the next command if the condition is TRUE
IF NOT	Executes the next command if the condition is FALSE
SHIFT	Shift positional parameters
START	Starts a program in the background

Chapter 6. Storage management

Storage management is the way disk space is managed by the operating system and how information is physically organized, written to, and read from the storage media. The way data is stored on disk is transparent to the end-user. However, this is probably the most critical aspect of an operating system. When information stored on disks is no longer available after a system malfunction or operator error, it is not at all transparent for the end-user. The reliability and performance of storage management is one of the keys to the success of an operating system.

This chapter describes how AIX and Windows 2000 manage their storage. Both the AIX file system and the Windows 2000 file system are described.

6.1 AIX storage management

AIX 5L storage management is based on the Logical Volume Manager (LVM).

6.1.1 Logical Volume Manager (LVM) terminology

The Logical Volume concept defines a higher-level interface that is transparent to applications and users and allows the flexible division, allocation, and management of fixed-disk storage space. This concept is implemented as a set of operating system commands, subroutines, device drivers, and tools that are collectively known as the Logical Volume Manager.

The basic LVM components and their relations are shown in Figure 36 on page 106.

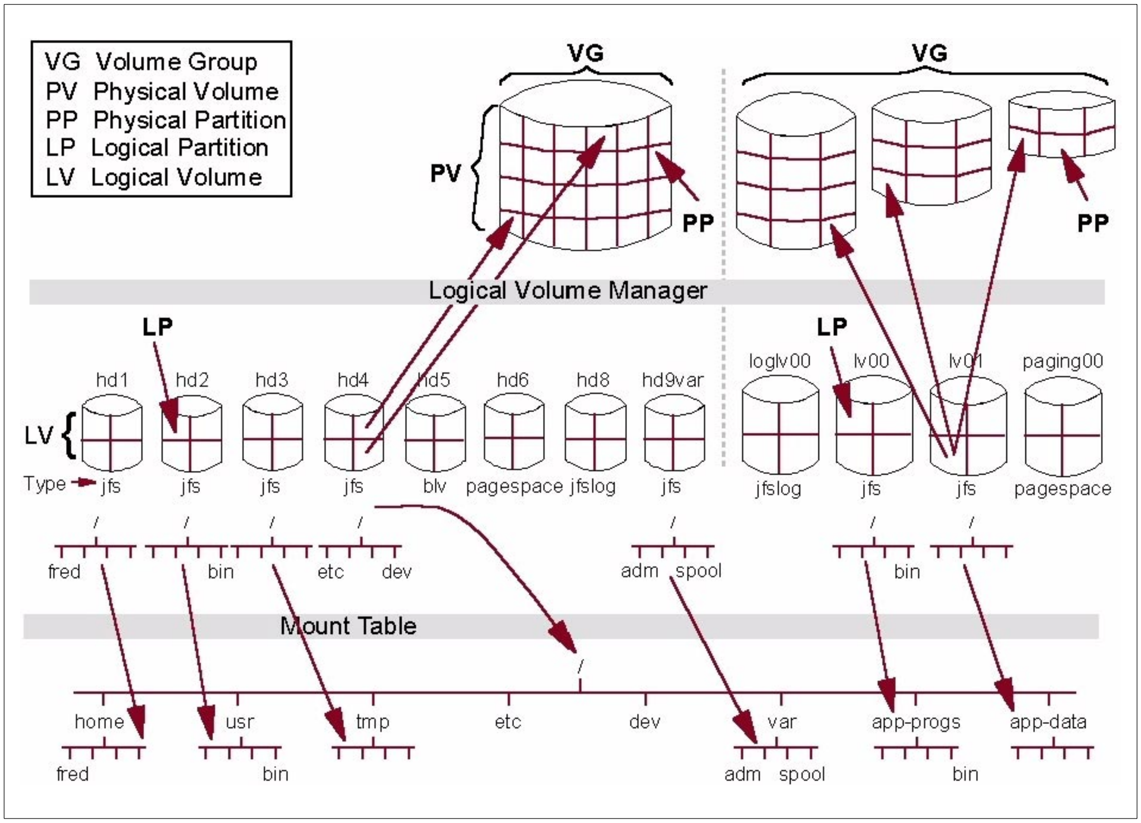


Figure 36. Components of the Logical Volume Manager

The basic components are:

6.1.1.1 Physical volume

The Physical volume is the physical disk drive and forms the basis of Logical Volume management. Before a physical disk can be used, it must be defined by the system. Each physical disk is assigned certain configuration and identification information that, together, define the disk as a Physical Volume (PV). This information is physically recorded on the disk and includes a Physical Volume Identifier (PVID) that uniquely identifies the disk. The disk is also assigned a Physical Volume name, typically, hdiskx, where x is a system unique number starting at 0. This Physical Volume name is also used for the low-level device driver interface to the disk, such as /dev/hdisk0.

Enhancement for physical volumes in AIX 5L

AIX 5L introduces the `-h` flag to the command `chvg` in order to specify a physical volume as a hot spare within a volume group. Once a hot spare has been defined, it can accept the migration of data from another physical volume that has begun to go bad. This enhancement can greatly increase the availability of important system and application data.

6.1.1.2 Volume Group

A Volume Group (VG) is a collection of between 1 and 32 Physical Volumes. In AIX 4.3.3, a new volume group format was added, which increased the maximum number of disks in Volume Group to 1024. Such a Volume Group is called Big Volume Group and cannot be imported to the systems with AIX Version 4.3.2 and below. Before Physical Volumes can be used by the system, they must be added to a Volume Group. A Physical Volume can only be a member of one Volume Group, but there can be up to 255 Volume Groups in a system.

Volume Group information includes a unique Volume Group Identifier (VGID), the PVIDs of all Physical Volumes in the Volume Group, and various kinds of status information. Each disk in the Volume Group has an area on disk known as the Volume Group Descriptor Area (VGDA) where this information is stored. The VGDA also contains information describing all of the Logical Volumes (discussed later in this section) that exist in the Volume Group. The size of the VGDA area limited the maximum numbers of disks to 32 in AIX prior to release 4.3.2. From this release on, the system administrator is able to configure Big Volume Group, which can contain up to 1024 disks. A special utility is provided to change a normal Volume Group to Big Volume Group. The limitations of Big Volume Groups are that rootvg cannot be converted to a Big Volume Group, a Big Volume Group cannot be accessed in *Concurrent* mode, and Big Volume Groups cannot be imported in systems with AIX releases lower than 4.3.2. If more than 128 Physical Volumes are attached to a system, more than one Volume Group will be required, and it is usually appropriate to design the system in such a way that different types of information are stored in different Volume Groups. For example, operating system information contained in one Volume Group and user information contained in another can assist in management and, in particular, recovery; should a disk fault occur in a Physical Volume from one Volume Group, only information from that Volume Group will be affected.

When you install AIX, one Volume Group is created by the installation process and is called rootvg. On this Volume Group, all System software is installed, primary paging space is defined, and the boot logical volume created.

6.1.1.3 Physical Partition

When a Physical Volume is added to a Volume Group, the space on the Physical Volume is divided up into equal chunks known as Physical Partitions (PPs). The PP size is set when a Volume Group is created, and all Physical Volumes that are added to the Volume Group inherit the same value. The Physical Partition size can range from 1 MB to 1024 MB and must be a power of 2, the AIX default being 4 MB. If you select 512 MB or 1024 MB for the size of PP, the Volume Group cannot be imported on systems with an AIX release prior to 4.3.1.

By default, up to 1016 Physical Partitions can be defined per Physical Volume. From AIX release 4.3.1 on, when you create a Volume Group you can specify the factor t, which enables more then 1016 PP per Physical Volume, but fewer PV per Volume Group (see Table 6).You can only specify factor t (with the option `-t factor`) from the command line (with the `mkvg` or `chvg` command), but not when using SMIT.

Table 6. Factor -t

Factor t	max.PP per PV	max. PVs in VG	max. PVs in VG - Big VG
1	1016	32	128
2	2032	16	64
3	3048	10	42
4	4064	8	32
5	5080	6	25
6	6096	5	21
7	7112	4	18
8	8128	4	16
16	16256	2	8

The Physical Partition is the smallest unit of disk space allocation in the Logical Volume paradigm. The Logical Volumes and file systems created on them can be defined in increments of the size of the Physical Partition. Smaller units increase allocation flexibility at the cost of increased management overhead.

6.1.1.4 Logical Partition

A Logical Partition (LP) is a pointer to 1, 2, or 3 Physical Partitions. The number of PPs the LP is mapped to is specified when a Logical Volume is created, but this can be changed later (increased or decreased). Information written to a Logical Partition will be physically written to the Physical Partitions pointed to. The number of Physical Partitions mapped to a Logical Partition defines the number of copies of that partition or the level of mirroring. Mirroring is usually done to two or three different PVs in order to increase the data availability. This means one LP is mapped to 2 or 3 PPs, each located on separate PVs (see Figure 36 on page 106).

Up to 32512 Logical Partitions can be defined per Logical Volume.

New AIX 5L logical partition enhancement

Using a new command introduced in AIX 5L called `lvmstat` you can monitor the performance of individual logical partitions to identify partitions that are getting more use than others. If disk usage on these partitions is less than efficient, you can use another new command in AIX 5L, `migratelp`, to split up logical partitions located on one disk over several disks to improve performance.

6.1.1.5 Logical Volume

Once a Volume Group has been created and Physical Volumes added to it, Logical Volumes can be created. A Logical Volume (LV) consists of a number of Logical Partitions and is an area of disk that can be used to store information.

The maximum number of user-definable Logical Volumes in a Volume Group is 256 for a normal Volume Group and 511 for a Big Volume Group.

Logical volumes are used to store such things as file systems, log volumes, paging space, boot data (Boot Logical Volume), and dump storage or can be used in Raw mode by databases, such as DB2 or Oracle. The section on Logical Partitions explained that a Logical Partition can be mapped to up to three Physical Partitions, which means that up to three copies of the information contained in a Logical Volume can be maintained. This is called mirroring.

A Logical Volume can have its size dynamically increased by adding Logical Partitions, the number of copies can be increased or reduced, and even the physical location of the Logical Volume on disk can be changed while the LV is in use. Figure 36 on page 106 shows the relationship between these components.

Since a Logical Volume can have up to 32512 Logical Partitions and since a Logical Partition size can be up to 1024 MB, the theoretical maximum size of a Logical Volume is 32512 GB or 31 TB (Terabytes). Because of the flexibility of system administration, the LVs are usually created with much smaller size.

6.1.2 Logical Volume Manager operation

This section describes the operation of the LVM.

6.1.2.1 General operation

As discussed previously, the Logical Volume Manager implementation consists of a set of operating system commands, library subroutines, and other tools that allow Logical Volumes to be established and controlled. These commands use the library subroutines to perform management and control tasks for the Logical Volumes, Physical Volumes, and Volume Groups in a system.

The interface to the Logical Volumes is called the Logical Volume Device Driver (LVDD), which is a pseudo device driver that manages and processes all I/O to Logical Volumes. The Logical Volume Device Driver is designed and utilized the same way as any other device driver in the system.

One of the responsibilities of the LVDD is mapping logical addresses to actual physical disk addresses, handling any mirroring, and maintaining Mirror Write Consistency (MWC). Mirror Write Consistency uses a cache in the device driver where blocks to be mirrored are stored until all copies have been updated. This ensures data consistency between mirrors. The lower half also manages bad block detection and relocation. If the physical disk is capable of this function, the Logical Volume Device Driver will make use of the hardware support; otherwise, it will be done in software. Both Mirror Write Consistency and bad block relocation can be disabled on a Logical Volume basis.

The list of data blocks to be written (or read) is finally passed by the Logical Volume Device Driver to the physical disk device drivers, which interact directly with the disks. In order for the Logical Volume Manager to work with a disk device driver, it must adhere to a number of criteria, the most significant of which is a fixed disk block size of 512 bytes.

The first 512 byte large block is reserved for the Logical Volume Control Block (LVCB). LVCB contains information about the creation date of the Logical Volume, information about mirrored copies of Logical Partitions, and information about the mount point, if there is a Journaled File System created on the Logical Volume. This is important to remember because a database application, such as DB2 or Oracle, using Raw Logical Volumes must not use the first 512 bytes because that is for the LVCB. If we do not do this, we will receive warning and error messages whenever AIX tries to read or update LVCB, which is virtually every time we make some changes on the Logical Volume. In AIX 4.3.3, a copy of LVCB is also held in the Volume Group Descriptor Area (VGDA).

6.1.2.2 Quorum

In order for a Volume Group to be accessible to the system, it must be *varied on*. During this process, the Logical Volume Manager reads management information from the Physical Volumes in the Volume Group. This information includes the Volume Group descriptor area already mentioned in Section 6.1.1, “Logical Volume Manager (LVM) terminology” on page 105, and another on-disk information repository known as the Volume Group Status Area (VGSA), which is also stored on all Physical Volumes in the Volume Group. The VGSA contains information regarding the state of Physical Partitions and volumes in the Volume Group, such as whether Physical Partitions are *stale* (used for mirroring, but not reflecting the latest information) and whether Physical Volumes are accessible. The VGDA is managed by the subroutine library, and the VGSA is maintained by the LVDD. If the `varyonvg` command cannot access a Physical Volume in the Volume Group, it will mark it as missing in the VGDA. For the command to succeed, a quorum of Physical Volumes must be available. A quorum is defined as a majority of VGDAs and VGSAs (more than 50 percent of the total number available). The only situation in which this is different is when there are only one or two Physical Volumes in a Volume Group. In this case, two VGDAs and VGSAs will be written to one disk and one (or none if there is only one disk) to the other. If the disk with two sets is inaccessible, a quorum will not be achieved and the vary on will fail. In this case, the system administrator will still be able to vary on VG with a -f flag (force flag). The Logical Volumes that reside completely or partially on missing Physical Volumes will not be accessible, but the ones on available Physical Volumes will, and backup of this data can be performed.

Note

If you want the Volume Group to stay varied on, even if it looses quorum, you must disable quorum. This is a common scenario, when you are implementing mirroring of rootvg. You configure one copy of LVs on first PV and the other copy of LVs on the second PV. In the case of one disk failure, you would usually want rootvg to be still varied on and thus available for the operating system. In this case, you must disable quorum by issuing the `chvg -Q n rootvg` command.

6.1.3 Logical Volume Manager policies

When Logical Volumes are created, there are a number of attributes that can be defined for them. These attributes govern the Logical Volume's operation regarding performance and availability. Attributes are policies that the LVM enforces for the Logical Volume. These policies are:

6.1.3.1 Intra-Physical Volume Allocation Policy

As shown in Figure 37, the Logical Volume Manager defines five concentric areas on a disk where physical partitions can be located. When creating a logical volume, the system administrator can choose a preferred disk location on which the physical partitions and, consequently, the logical partition that will make this logical volume should be created. This can be used to optimize overall logical volume and file system performance. The characteristics of these regions are described below:

- **Edge** and **Inner Edge** - These regions, generally, have the longest disk arm seek times resulting in the slowest average access times. Logical Volumes containing least frequently accessed data are best located here.
- **Outer Middle** and **Inner Middle** - These regions provide better average seek times than Edge and Inner Edge and, consequently, lower average access times. Reasonably frequently accessed data should be positioned here.
- **Center** - This region provides the lowest average seek times and, thus, the best response times. Information that is accessed regularly and needs high performance should be situated here.

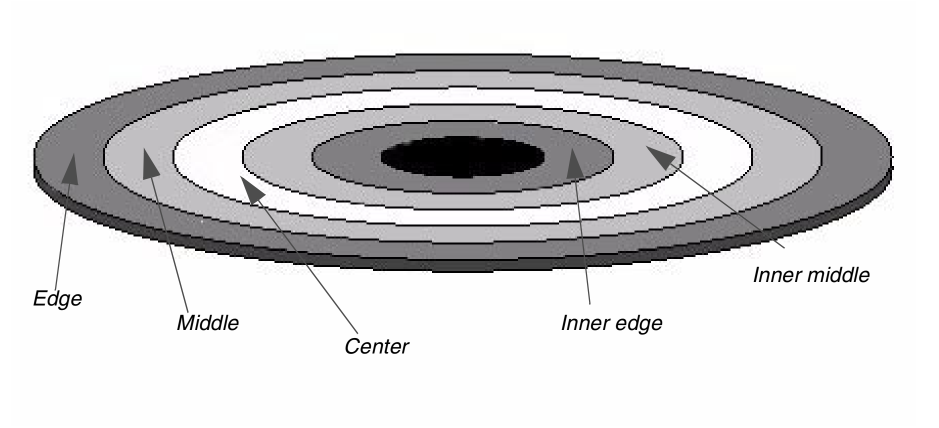


Figure 37. Physical disk partition location - Intra-disk allocation policy

The different average seek times are based upon the supposition that there is a uniform distribution of disk I/O, meaning the disk head will spend more time crossing the center section of the disk than any of the other regions.

When creating a logical volume, the Logical Volume Manager will do its best to locate the logical volume as closely as possible to the required position. If, however, there is no space available in the required area, LVM will create LV on another position with no warning message returned to the system administrator. To confirm that LV was created in a designated area, the system administrator can issue the following command:

```
# lslv -l hd2
```

The result would be similar to the following:

```
hd2:/usr
PV              COPIES        IN BAND       DISTRIBUTION
hdisk0          154:000:000   61%           000:033:095:026:000
```

In this example, 95 PPs of Logical Volume, hd2, are located in the Center region; 33 PPs are located in the Inner Middle, and 26 PPs are located in the Middle region.

6.1.3.2 Mirroring

A logical volume is made of logical partitions. The Logical Volume Manager allows each logical partition in a logical volume to be mapped to one, two, or three physical partitions. This means that up to two copies of a logical volume can be transparently maintained for performance and availability purposes.

The scheduling policy explained below determines how information is actually written to disk when mirroring is used. Should a disk with one of the copies of the logical volume fail, or should some of the physical partitions in the copy become damaged, another copy can be transparently used while repairs are effected. Furthermore, the copy that has the required partitions closest to a read/write head will be used for reading, thus, improving performance. The benefits here are somewhat dependent upon the inter-physical volume allocation policy, which is explained next.

When mirroring is being used, there are two ways in which the Logical Volume Manager can schedule I/O for the physical volumes:

- **Sequential-write copy** - When this option is selected for a logical volume, write requests are performed for each copy successively in the following order: Primary, secondary, and tertiary. A write to a copy must complete before the next copy can be updated, thus ensuring maximum availability in the event of failure.

Read requests will be initially directed to the primary copy. If this fails, they will be directed to the secondary, and then to the tertiary if necessary (and defined). While the data is being read from the next copy, the failing copy or copies are repaired by turning the read into a write with bad-block relocation switched on.

- **Parallel-write copy** - In this case, write requests are scheduled for each of the copies simultaneously. The write request returns when the copy that takes the longest to update completes. This method provides the best performance.

  Read requests are scheduled to the copy that can be most rapidly accessed, thereby minimizing response time. If the read fails, repairs are accomplished using the same mechanism as for sequential-write copy.

If required, there is a great deal of additional information, in the form of online documentation, about all aspects of the Logical Volume Manager.

Inter-disk physical volume allocation policy

When the Logical Volume Manager allocates partitions for a logical volume, the partitions can be spread across multiple disks. The inter-physical volume allocation policy governs how this will actually be implemented in terms of numbers of physical volumes. There are two options:

- **Minimum** - The minimum option indicates that, if mirroring is being used, the minimum number of physical volumes should be used per copy, and each copy should use separate physical volumes. If mirroring is not being used, just the minimum number of physical volumes necessary to hold all of the required physical partitions should be used.
- **Maximum** - The maximum option attempts to spread the required physical partitions over as many physical volumes as possible, thereby improving performance. If mirroring is not used here, this approach is highly-sensitive to physical volume failure. The loss of any physical disk will result in the loss of the logical volume.

6.1.3.3 Striping

When creating a logical volume, it is also possible to define a stripe size if you want the logical volume to be striped. The stripe size can be 4 KB, 8 KB, 16 KB, 32 KB, 64 KB, or 128 KB.

AIX 5L allows the placement of logical volumes on a specific area of one or more physical volumes. For example, the center of the disk may be chosen for the placement of logical volumes when rapid access to data is required. Even though this placement strategy can provide fast access to data, it is still restricted by the fact that a disk I/O operation is performed to retrieve each data block.

In Part 1 of Figure 38, the numbered disk blocks for the file represent the sequence of data in the file. To read the entire file sequentially involves reading each disk block in turn.

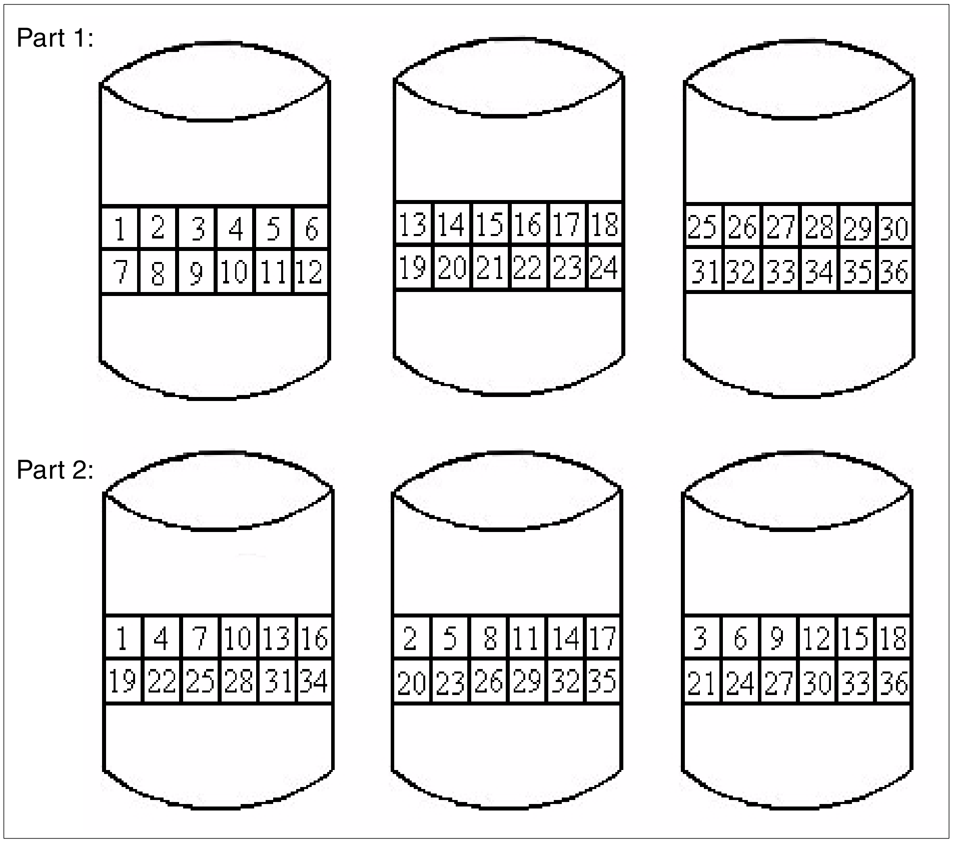


Figure 38. Striping example

However, if we place the data in a logical volume over all available disks to enable parallel access to that data, this would further improve sequential access to that data (see Figure 38).

In user environments where sequential access to large data files is very frequent, this technique proves to be extremely efficient. In fact, AIX 5L provides this technique with a mechanism known as striping. In non-striped logical volumes, data is accessed using addresses to data blocks within physical partitions. In a striped logical volume, data is accessed using addresses to stripe units. Consecutive stripe units are created on different physical volumes.

A single stripe consists of a stripe unit on each physical volume. The size of a stripe unit must be specified at creation time and can be any power of 2 in the range 4 KB to 128 KB. Because data in a striped logical volume is no longer accessed using data block addresses, the LVM will track which blocks on which physical drives actually hold the data being accessed. If the data being accessed resides on more than one physical volume, the appropriate number of simultaneous disk I/O operations will be scheduled for all drives concerned.

Mirroring and striping support

Starting in AIX 4.3.3, you have the ability to create logical volumes that are mirrored and striped, which provides you with high-performance striped logical volumes and high availability of mirrored logical volumes, see Figure 39 on page 117.

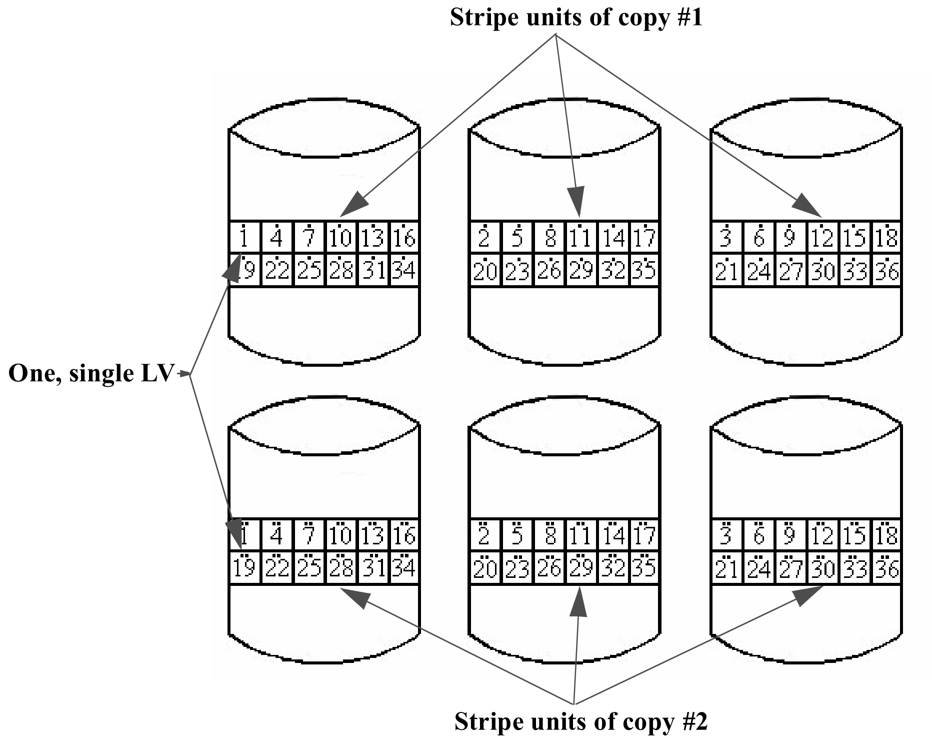


Figure 39. Striped and mirrored logical volume

For striped and mirrored logical volumes, a new allocation policy is introduced: *Super strict*. This allocation policy ensures that the partitions allocated for one mirror cannot share a physical volume with the partitions from another mirror. If you are creating a striped and mirrored logical volume with SMIT, this policy will be selected automatically. However, if you are creating a logical volume with the `mklv` command, you should specify the super strict allocation policy with the `-s` flag.

6.1.3.4 Bad-block relocation

As was mentioned in “General operation” on page 110, the Logical Volume Manager will perform bad-block relocation if required. This is the process of redirecting read/write requests from a disk block that has become damaged to one that is functional; this happens transparently to an application.

6.1.3.5 Concurrent online mirror backup

Starting in AIX 4.3, AIX provides support for online backup mechanism for a mirrored logical volume. The idea is to make one copy of the logical volume temporarily inactive for applications; it means that any updates of LV will take place only on one copy, and, in this time, perform a backup of the logical volume from the non-active copy.

6.1.4 Logical Volume Manager benefits

The LVM provides the following benefits:

- Transparent control of physical storage data contained in a logical volume appears to be contiguous, but it can be located on disk partitions that are not side-by-side, or even on the same physical disk. This allows efficient use of available disk space, particularly when logical volumes require expansion.
- Mirrored copies of logical volumes, by being able to assign multiple physical partitions to each logical partition, enable copies of vital information to be transparently maintained (even on separate physical disks) for additional security.
- Striped logical volumes enable many simultaneous reads from many physical volumes for applications with high-performance sequential read requirements. From AIX 4.3.3 on, striped logical volumes can also be mirrored.
- Starting in AIX 5L you can designate a disk within a volume group as a hot spare. In the event of a disk failure a hot spare can take over, and AIX 5L will begin to migrate what data is still good to the designated hot spare.
- Capacity greater than physical disk sizes - The logical partitions comprising a logical volume can span multiple disks, which means that logical volumes are not limited to the sizes of the individual physical disks attached to the system.
- Physical partition flexibility - The sizes of physical partitions can be defined when a volume group is created. This gives flexibility in the use of disk resources. For example, logical volumes can be increased in size by smaller increments, thereby, utilizing the available disk space more effectively.

- Logical partition flexibility - starting in AIX 5L logical partitions can be migrated from one physical partition to another, allowing system administrators to customize their storage management for better performance.

6.1.5 AIX 5L file systems

One further level of abstraction is provided at the operating system level, and this is the file system. A file system is, essentially, a hierarchical structure of directories, each directory containing files or further directories (subdirectories). The main purpose of a file system is to provide for improved management of data by allowing different types of information to be organized and maintained separately. However, as will be shown later in this section, file systems also provide many more facilities.

There many different types of file systems in existence, including the following:

AIX Journaled File System (JFS)
This is the native AIX file system providing the full range of supported file system operations for organizing and managing physical files. The JFS is explored in more detail later in this section. The JFS is created within a logical volume.

On all UNIX systems, there must be at least one file system, called the root file system or /, within which the other file systems can be accessed on the local system. In AIX 5L, several other file systems are created at installation time: /usr, /var, /tmp, /home, /opt, and /proc.

The /proc file system
New in AIX 5L, the /proc file system contains a directory for each kernel data structure and active process running on the system.

Each of these entries gets a Process Identification Number (PID) within kernel memory, and now within AIX 5L each PID gets its own directory structure within /proc.

Working with kernel data structures and processes in this manner allows a debugger or system administrator to stop and start threads within a process, trace syscalls, trace signals, and read and write to virtual memory within a process. The new /proc file system can be invaluable in debugging system processes and applications.

The /opt file system

Also new to AIX in 5L is the /opt, or “optional” directory. This directory is reserved for the installation of add-on application software packages, and is integral to AIX 5L’s new affinity with Linux applications.

6.1.5.1 JFS2 in AIX 5L

Based on standard JFS from AIX 4.3.3, the Journaled File System 2 (JFS2) is an enhanced and updated version intended to provide a more robust and functionally thorough file system implementation for AIX 5L systems.

JFS2 has the following advantages over standard JFS:

- File and file system size increased to 4 petabytes (PB) (both only tested up to one terabyte)
- Keeping record of i-nodes is now dynamic and the number of I-nodes in a file system is limited only to disk size
- Directory structures supported in JFS2 include B-tree and linear (standard JFS only supports linear directories)
- Improved consistency and recover-ability of journaled file systems
- Quicker restart times for files systems using log-based journaling of files
- Defragmentation is now capable on a mounted and actively accessed file system

Furthermore, JFS2 is, in certain scenarios, compatible with standard JFS. While an AIX 4.3.x file system cannot import a JFS2 file system, an AIX 5L system can import file systems from AIX 4.3.x into JFS2. However, with NFS there is full compatibility and there are no issues with an AIX 4.3.x system mounting a remote JFS2 file system or an AIX 5L system mounting a remote standard JFS file system. NFS is described in more detail in Section 6.1.5.2, “Network File System (NFS)” on page 121.

Additionally, SMIT has been upgraded to support JFS2. You can use the fast path `smit jfs2` to create a JFS2 file system using SMIT. Also, the Web-based System Manager has been upgraded to support JFS2, and the AIX commands `crfs`, `fsck`, and `logform` have all been enhanced to support JFS2. The command `crfs` is used to create a file system, `fsck` checks file systems for errors, and `logform` is implemented to prepare a logical volume for use as a JFS or, now, a JFS2 log.

6.1.5.2 Network File System (NFS)

NFS allows files and directories located on other systems over a TCP/IP network to be incorporated into a local file system and accessed as though they were a part of that file system. NFS provides its services on a client/server basis. Server systems can make selected files and directories available for access by client systems.

NFS provides a number of services including the following:

- **Mount Service** - This service allows clients to mount into a local file system the portion of the remote file system that they wish to access. Mounting is discussed in more detail later in this section.
- **Remote File Access** - This service fulfills requests for file activity from the client to server (such as opens, reads, and writes).
- **Remote Execution Service** - This service allows authorized clients to execute commands on the server.
- **Remote System Statistics Service** - This service provides statistics on the recent availability of the server.
- **Remote User Listing Service** - This service provides information to clients about users of the server system.

NFS installation, configuration, and management are covered in detail in the online documentation at `http://www.rs6000.ibm.com/doc_link/en_US/a_doc_lib/aixbman/commadmn/ch10_nfs.htm`

NFS operation is stateless, which means that the server does not maintain any transaction information on behalf of clients. Each file operation is atomic, which means that no information on the operation is retained. Thus, if a connection should fail, it is up to the client to maintain any synchronization or transaction logging to ensure consistency.

6.1.5.3 CD-ROM File System (CDFS)

This type of file system allows the contents of a CD-ROM to be accessed as if they were part of a local file system.

The following standard CD-ROM formats are supported:

- **The ISO 9660:1988(E) standard** - The CDRFS supports ISO 9660 level 3 of interchange and level 1 of implementation.
- **The High Sierra Group Specification** - Precedes the ISO 9660 and provides backward compatibility with previous CD-ROMs.

- **The Rock Ridge Group Protocol** - Specifies extensions to the ISO 9660 that are fully-compliant with the ISO 9660 standard and provide full POSIX file system semantics based on the System Use Sharing Protocol (SUSP) and the Rock Ridge Interchange Protocol (RRIP) enabling mount/access CD-ROM as with any other UNIX file system.
- **The CD-ROM eXtended Architecture File Format (in Mode 2 Form 1 sector format only)** - This file format specifies extensions to the ISO 9660 that are used in CD-ROM-based multimedia applications, such as Photo CD.

6.1.5.4 Andrew file system

The Andrew File System (AFS), provides a similar basic service to NFS in that it allows machines to access remote file systems as though they were local. The major difference is that AFS defines its own hierarchy, one where many machines can participate in mounting sections of their local file systems into the hierarchy. Client machines that are authorized can then mount the entire AFS hierarchy into their local file system structure and, thereby, access information on a wide range of machines and file systems as though it were local.

6.1.5.5 OSF distributed file system (DFS)

This file system is part of the OSF Distributed Computing Environment (OSF DCE). It provides file-sharing capabilities with a high level of availability and security.

6.1.5.6 AIX 5L standard journaled file system implementation

The AIX Journaled File System (JFS) is implemented through a set of operating system commands that allow the creation, management, and deletion of files and a set of subroutines that allow lower-level access, such as open, read, write, and close, to files in the file system.

A JFS is created on a logical volume and organized as shown in Figure 40 on page 123.

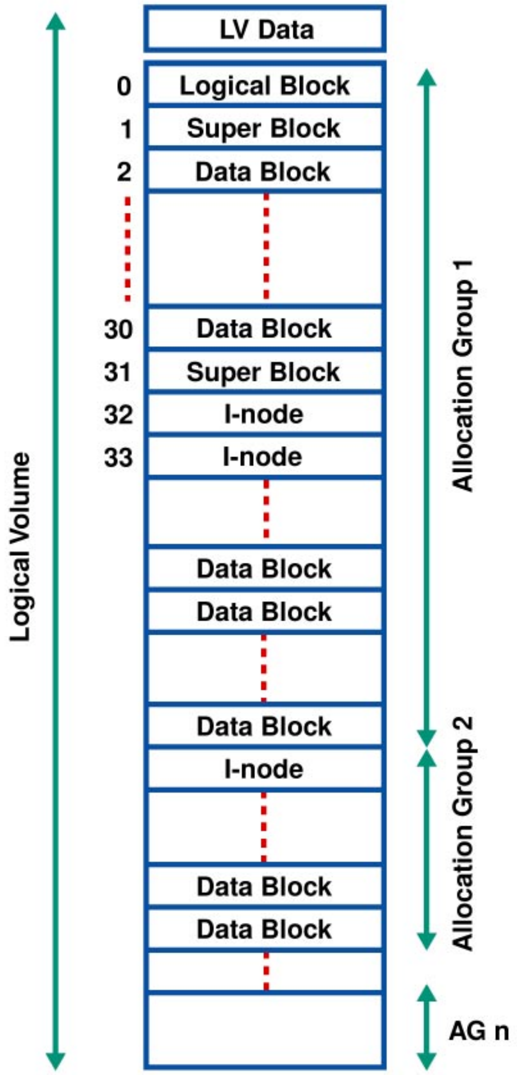


Figure 40. JFS physical organization on a logical volume

As can be seen, the JFS divides the logical volume into a number of units or logical blocks of fixed size. Before AIX V4, these logical blocks had a fixed-size of 4096 corresponding to the memory page size.

While, generally, efficient from the point of view of loading the file into memory and preventing physical disk fragmentation, having a fixed logical block size can have drawbacks. If the majority of files stored in the file system are small (less than one logical block in size), there will be a great deal of wasted disk space in the accumulation of those portions of the logical blocks that remain unused by the smaller files. For example, if all files are less than half of a logical block in size, half of the total file system space will be unused even though the file system is full.

AIX V4 has introduced fragments in the structure of a file system. A fragment is the smallest unit of file system disk space allocation. The fragment size can be set at the creation of the file system and is stored in the superblock. The size can be 512, 1024, 2048, or 4096 bytes. This fragment size is used for I/O at the file system interface. This means that the file system passes data to be written to the LVM or receives data that has been read from the LVM using blocks of data having the fragment size set at file system creation.

The logical blocks in the file system are organized as follows:

Logical Block 0
The first logical block in the file system is reserved and available for a bootstrap program or any other required information; this block is unused by the file system.

Superblock
The first and thirty-first logical blocks are reserved for the superblock (logical block 31 is a backup copy). The superblock contains information, such as the overall size of the file system in 512-byte blocks, file system name, file system log device (logs will be covered later in this section), version number, and file system state.

Allocation Groups
The rest of the logical blocks in the file system are divided into a number of allocation groups. An allocation group consists of data blocks and i-nodes to reference those data blocks when they are allocated to directories or files. The reasons for this extra level of abstraction are:

- **To improve locality of reference** - Files created within a directory will be maintained in an allocation group with that directory. Since allocation groups consist of contiguous logical blocks, this should assist in maintaining locality of reference for the disk head.
- **To make file system extension easier** - Extending a file system is easier because a new allocation group of i-nodes and data blocks can be added maintaining the relationship between i-nodes and file system size simply.

> Without allocation groups, the file system would either have to be reorganized to increase the number of i-nodes, or the extension could only increase the number of data blocks available, thereby, conceivably, limiting the number of files and directories in the file system.

I-nodes are explained next. For a better representation of this organization, refer to Figure 40 on page 123.

i-nodes

Basically, an i-node is a pointer to a file. An i-node contains information about the file, such as the type of file, the size in bytes, the owner, access permissions for the user ID (UID) and the group ID (GID), the number of blocks allocated to the file, the creation date, last modification date, last access date, and pointers to the blocks that actually contain the file.

When the file system is created, a certain number of blocks are reserved for storing the i-nodes. The number of i-nodes determines the number of files or directories that can be created on that file system. By default, the number of i-nodes is such that each 4096 byte block of the file system can be addressed and contain a file.

This is not the most efficient way of using the space in the file system; so, in AIX V4, we have the possibility of specifying the number of bytes per i-node (NBPI).This number specifies the ratio of the file system size (in bytes) to the number of i-nodes. For example, if the NBPI is set to 16384, in a file system of size 32 MB, 2000 i-nodes will be created. This means it will be possible to create 2000 files and directories in that file system. The NBPI can be from 512 to 131072, and the default is 4096.

These i-nodes will be divided between the allocation groups.

The structure of an i-node is described in Figure 41 on page 126, where an NBPI of 4 KB is assumed. The first part contains information, such as the owner, creation and modification dates, and permissions for the directory or file. The second part contains an array of eight pointers to the actual disk addresses of the 4 KB logical blocks that make up the file or directory.

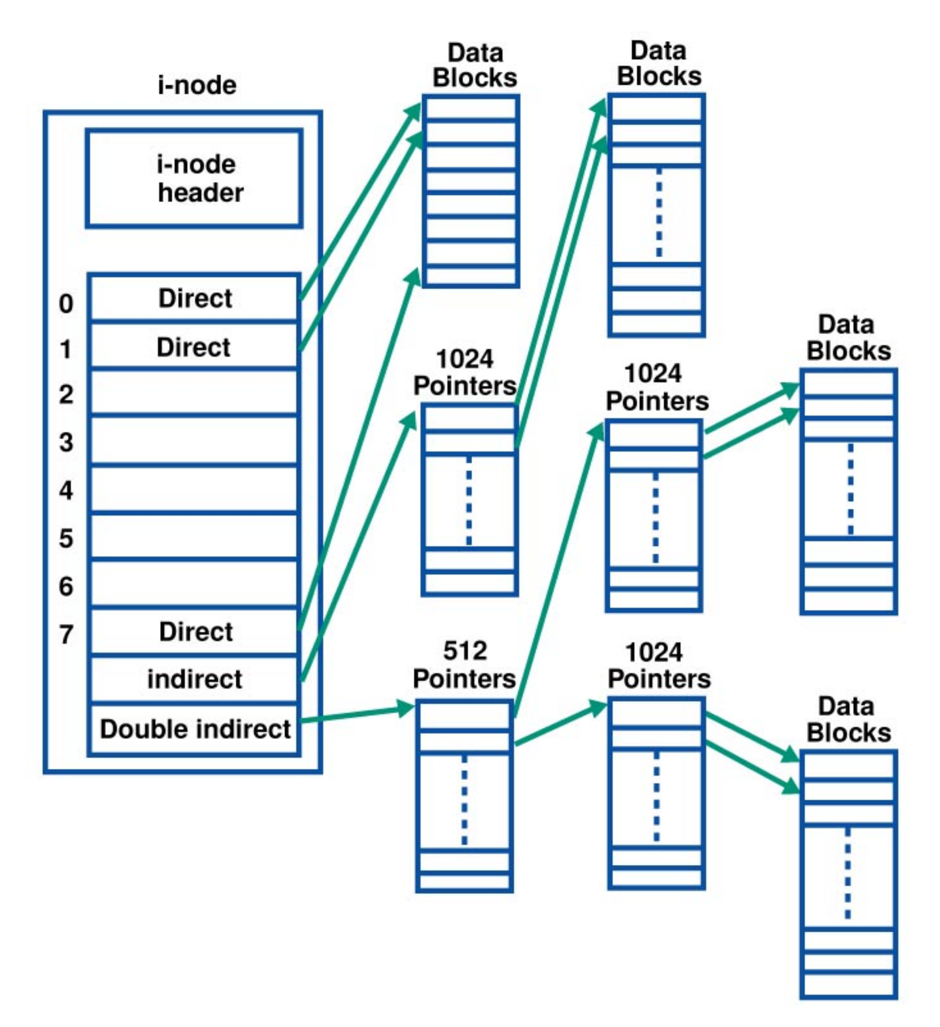


Figure 41. Structure of an i-node

For files that can fit within the array storage area, such as most links, the file is actually stored in the i-node itself, thus, saving disk space.

For a file of up to 32 KB (8 x 4 KB) in size, each i-node pointer will directly reference a logical block on the disk. For example, if the file has a 27 KB size, the first seven pointers will be required, the last pointer referencing a 4 KB logical block containing the last 3 KB of the file.

For files up to 4 MB, the i-node points to a logical block that contains 1024 pointers to logical blocks that will contain the files data; This gives a file size of up to 1024 x 4096 or 4 MB.

For files greater in size than this, the i-node points to a logical block that contains 512 pointers to logical blocks each containing 1024 pointers to the logical blocks that will actually contain the file's data. This gives a maximum file size of 512 x 1024 x 4096 or 2 GB.

6.1.5.7 Large-file-enabled Journaled File System

If files bigger than 2 GB are required, a large-file-enabled Journaled File System must be created.

The geometry for a large-file-enabled JFS allows some of the indirect blocks to contain disk addresses that refer to larger blocks. Specifically, the entries in the first indirect block point to normal 4 KB data blocks, and all of the entries in other indirect blocks point to larger (128 KB) data blocks (see Figure 42 on page 128)

In file systems enabled for large files, file data stored before the 4 MB file offset is allocated in 4096-byte blocks. File data stored beyond the 4 MB file offset is allocated with large disk blocks 128 KB in size. The large disk blocks are actually 32 contiguous 4096-byte blocks.

For example, a 132 MB file in a file system enabled for large files uses two single indirect blocks (one with 1024 x 4 KB data blocks and one with 1024 x 128 KB data blocks). In a regular file system, the 132 MB file would require 33 single indirect blocks (each filled with 1024 x 4 KB disk blocks).

The maximum file size for this geometry is: (1 x 1024 x 4 K) + (511 x 1024 x 128 K) or 68589453312 Bytes (or around 64 GB).

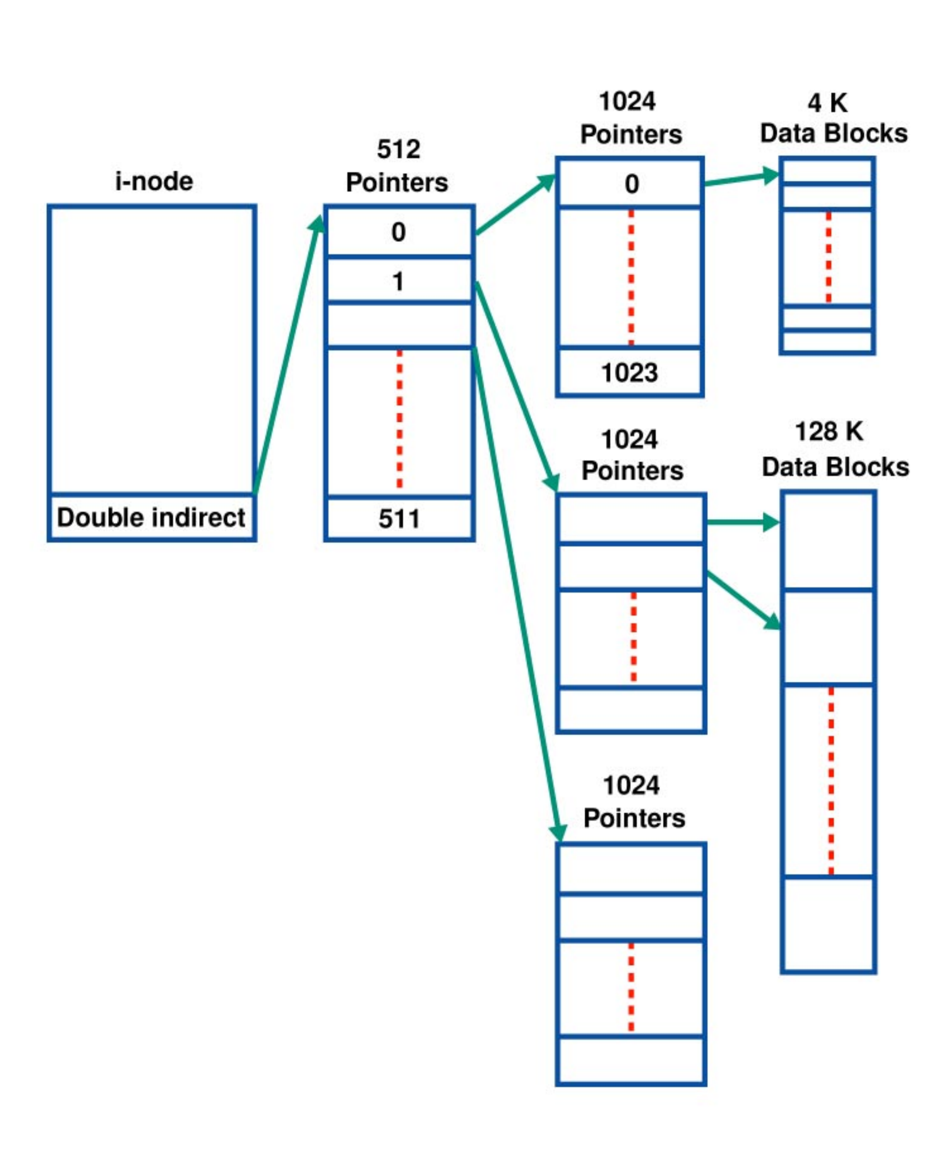


Figure 42. JFS geometry for large files

6.1.5.8 Compressed Journaled File System

AIX V4.3 allows the system administrator to create a compressed file system. This facility provides compression of regular files (as opposed to directories or links). The compression is implemented on a logical block basis, which means that when a logical block of file data is to be written, an entire logical block is allocated for it; the logical block is then compressed, and the number of fragments now required as a result of the compression is actually allocated. Thus, in contrast to a fragment file system, which only allocates fragments for the final logical blocks of files less than 32 KB in size, compressed file systems allocate fragments for every logical block in every file.

Compression is done block-by-block in order to fulfill the requirements for efficient random I/O. The algorithm used is LZ1.

If a compressed file system is used, there is the potential for file system fragmentation. Although enough free fragments are available for writing a new file, they are not in contiguous space and cannot be used. In order to solve this problem, there is a utility called defragfs, which can be used for defragmenting a file system.

Note

The / (root) and /usr file systems must not be compressed!

6.1.5.9 Journaled File System recoverability

Since AIX V3.1 came out in 1990, the AIX JFS has been a recoverable file system. All transactions issued on the file system structure (file system meta data) are logged into a JFS log logical volume. This transaction log provides file system recovery in case the system abnormally terminates.

By default, one JFS log, with a default size of one logical partition, maintains log data for all the file systems within a volume group. The default log size may be increased for file systems that are larger than 2 GB. The maximum size for a JFS log is 256 MB.

It is also possible to create a separate JFS log logical volume for a specific file system. Since the JFS log is a logical volume, the physical location of the log can be set at log logical volume creation. This is very important in terms of performance. Having the log on a physical volume other than the file system might improve write access to this file system.

6.1.5.10 Journaled FIle System limitations

The JFS is designed to support up to 2^{24}, or 16 million, i-nodes. With an NBPI (Number of Bytes Per i-node) of 512, this leads to 512 x 16 M = 8 GB.

The theoretical file system size limit is reached with an NBPI of 131072, which gives a maximum file system size of 131072 x 16 M = 2097152 MB, that is, 2 TB (2 Terabytes). Since this size requires very large hard drives, it was not possible to test such a size; therefore, IBM announced a maximum supported file size of 1024 GB. See Table 7.

Table 7. AIX 5L Journaled File System maximum size

NBPI	Minimum Allocation Group Size (MB)	Fragment Size (Bytes)	Maximum File System Size (GB)
512	8	512,1024,2048,4096	8
1024	8	512,1024,2048,4096	16
2048	8	512,1024,2048,4096	32
4096	8	512,1024,2048,4096	64
8192	8	512,1024,2048,4096	128
16384	8	512,1024,2048,4096	256
32768	16	1024,2048,4096	512
65536	32	2048,4096	1024
131072	64	4096	1024

6.1.5.11 Journaled File System Long File Name Support

The AIX JFS has supported long filenames since AIX became available in 1990. File names can have up to 255 characters.

6.1.5.12 On-line Journaled File System backup

In AIX 4.3.3, you also have the ability to make an online backup of JFS, not just raw logical volumes. In order to do this, the logical volume on which JFS resides must be mirrored as must its JFS log. When doing an online backup of JFS, LVM creates a snapshot of the logical volume on which JFS resides. However, because of asynchronous file writes, the snapshot may not contain all the data that was written immediately before the snapshot was taken. Therefore, it is recommended that file system activity be minimal while the snapshot (the splitting of logical volume copies) is taking place. After the backup is finished, the copy of the LV (JFS) that was used for backup is reintegrated and synchronized with the primary copy.

6.1.5.13 Accessing file systems

Once a file system of whatever type has been created, it must be mounted in order to access the information within it. The process of mounting creates the connection between an existing and accessible local file system mount point and the root directory of the directory structure to be accessed. A mount point can be either a directory or a file in the local file system. If a new local file system or remote directory structure (using NFS for example) is to be accessed, it must be mounted over a directory. If only a single file is to be accessed, it must be mounted over a local file. The AIX operating system starts with a root file system into which the /usr, /var, /tmp, and /home file systems are mounted at boot time. Any other required file systems can then be mounted wherever they are needed (assuming relevant permissions).

6.1.6 AIX 5L and RAID support

In order to improve the availability of data, several different concepts of configuring disks are used by many vendors of storage products. Apart from disk, these configurations usually use special disk controller and/or special controller software. Different configurations of disks are called RAID levels. AIX supports RAID levels 0, 0+1, and 1 as part of Logical Volume Manager. If the customer wants to implement RAID level 5, a special disk controller should be used, such as IBM Ultra SCSI RAID controllers (2493, 2494) or IBM SSA RAID controllers (6225).

RAID 0

RAID 0 is also known as data striping. Conventionally, a file is written out to (or read from) a disk in blocks of data. With striping, the information is split into chunks (a fixed amount of data), and the chunks are written to (or read from) a series of disks in parallel.

RAID 0 is well suited for program libraries requiring rapid loading of large tables or, more generally, for applications requiring fast access to read-only data or for fast writing. RAID 0 is only designed to increase performance; there is no redundancy. Therefore, any disk failures will require reloading from backups.

AIX JFS supports RAID 0 with the creation of a striped logical volume.

RAID 1

RAID 1 is also known as disk mirroring. In this implementation, duplicate copies of each chunk of data are kept on separate disks or, more commonly, each disk has a twin that contains an exact replica (or mirror image) of the information. If any disk in the array fails, the mirrored twin can take over. Read performance can be enhanced because the disk with its actuator closest to the required data is always used, thereby, minimizing seek times. RAID 1 is best suited to applications requiring high data availability and good read response times and where cost is a secondary issue.

AIX JFS supports RAID 1 with the possibility of having up to two copies of the logical volume on separate physical volumes.

RAID 0+1

This is the combination of data striping and disk mirroring described above.

AIX 5L supports RAID 0+1. See “Mirroring and striping support” on page 116, for more details.

RAID 2/RAID 3

These are parallel process arrays where all drives in the array operate in unison. Similar to data striping, information to be written to disk is split into chunks, and each chunk is written out to the same physical position on separate disks (in parallel). When a read occurs, simultaneous requests for the data can be sent to each disk. Then, each request retrieves the data from the same place and returns it for assembly and presentation to the requesting application. More advanced versions of RAID 2 and 3 synchronize the disk spindles so that the reads and writes can truly occur simultaneously (minimizing rotational latency buildups between disks). This architecture requires parity information to be written for each stripe of data. The difference between RAID 2 and RAID 3 is that RAID 2 can utilize multiple disk drives for parity, while RAID 3 uses only one. RAID 2 is rarely used, but RAID 3 is well suited for large data objects, such as CAD/CAM or image files or for applications requiring sequential access to large data files.

AIX JFS needs specific hardware to support RAID 2/3.

RAID 4

RAID 4 addresses some of the disadvantages of RAID 3 by using larger chunks of data and striping the data across all of the drives except the one reserved for parity. Using disk striping means that I/O requests need only reference the drive on which the required data actually resides. This means that simultaneous, as well as independent, reads are possible. Write requests, however, require a read/modify/update cycle that creates a bottleneck at the single parity drive. This bottleneck means that RAID 4 is not used as often as RAID 5, which implements the same process but without the bottleneck.

AIX JFS needs specific hardware to support RAID 4.

RAID 5

As has been mentioned, RAID 5 is very similar to RAID 4. The difference is that the parity information is distributed across the same disks used for the data, thereby, eliminating the bottleneck. Parity data is never stored on the same drive as the chunk that it protects. This means that concurrent read and write operations can now be performed, and there are performance increases due to the availability of an extra disk (the disk previously used for parity). There are other enhancements possible to further increase data transfer rates, such as caching simultaneous reads from the disks, and then transferring that information while reading the next blocks. This can generate data transfer rates at up to the adapter speed. Similar to RAID 3, in the event of disk failure, the information can be rebuilt from the remaining drives. RAID 5 is best used in environments requiring high availability and fewer writes than reads.

AIX 5L JFS and JFS2 require specific hardware to support RAID 5.

6.2 Windows 2000 storage management

This section describes how Windows 2000 manages its disk space and which fault-tolerant functions are provided in the operating system itself.

6.2.1 Volume management

In Windows 2000, there are two types of disks: Basic disks and Dynamic disks. Basic disks are configured using the partitions inherited from Windows NT and earlier releases of DOS/Windows. Dynamic disks can be used with the new part of the operating system called the Logical Disk Manager (LDM), which is shown in Figure 43 on page 134. The basic entity in Windows 2000 storage management is called a *Volume*.

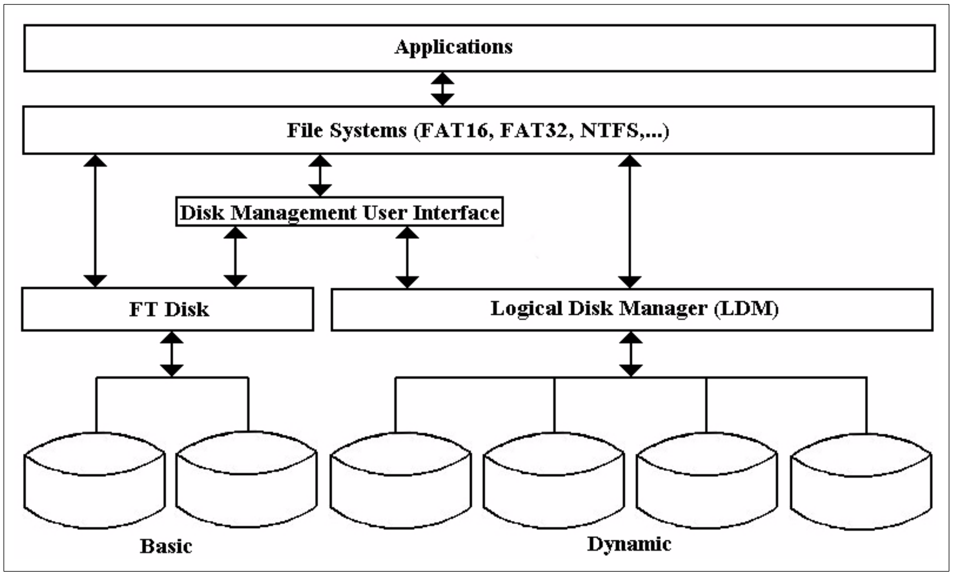

Applications
File Systems (FAT16, FAT32, NTFS,...)
Disk Management User Interface
FT Disk
Logical Disk Manager (LDM)
Basic
Dynamic

Figure 43. Windows 2000 Volume Management Concept

6.2.1.1 Volume management terminology

The following sections describe the volume management terminology.

Basic disk

Basic disks have the same structure as in Windows NT, storing their configuration information in the Master Boot Record (MBR) where all information about partitions and partition sets is stored. Some additional information, added by Windows 2000, is stored on the first track of the disk.

Disk group

A disk group is a collection of dynamic disks that are managed as a collection. Each disk in a disk group has a specific area (1 MB in size) that stores the configuration data about this disk group (similar to VGDA in AIX).

Dynamic disk

Dynamic disks are part of the disk group. Dynamic disks are managed by the Logical Disk Manager, and simple, spanned, striped, or distributed parity (RAID 5) volumes can be created on them using software functions. After upgrading from Windows NT to Windows 2000, all existing disks are recognized as basic disks but can be converted to dynamic disks.

Partition

A partition is the part of the disk where logical volumes are created. Each basic disk can be divided into several partitions. They can be of two types: Primary and extended. A primary partition can be marked as bootable and at least one must be available in the system. A logical drive is associated when the partition is created and is identified with a letter, such as C.

Volume

A volume is the part of the disk that is formatted with a file system.

On Basic disks, you can still use simple partitions, extended partitions, or fault-tolerant sets. These partitions are managed by the Ftdisk device driver in the same way as in Windows NT. There are several different kinds of volumes:

- **Simple volume** - Configured on the free area of a single disk, it can occupy one contiguous area or consist of two or more concatenated areas of the disk. It can be extended to other disks, and is then referred to as a *spanned volume*.
- **Striped volume (RAID 0)** - Consecutive stripes of data are written to different disks (interleaved) in order to increase read and write performance.
- **Mirrored volume (RAID 1)** - This is a fault tolerant volume where two copies of the data exist on two different disks. If one of the disks fail, the data can still be read and written to the copy of the volume on the other disk.
- **Spanned volume** - This volume can be configured to span up to 32 physical disks.
- **RAID 5 (distributed parity) volume** - This is a fault tolerant volume where the data is striped across an array of three or more physical disks. Special chunks of data, called *Parity* (the calculated value of the chunk of data made with a logical XOR operation), are also striped across the disks. If one of the disks fails, the original data can still be recreated from the other disks and the parity information.

Volumes, just as partitions, can be associated with a letter, or, similar to AIX, be mounted over a directory. This functionality is new to Windows 2000 and is enabled with the concept of *reparse points*. A reparse point is a special tag that is applied to a file system or directory and passed back to the driver stack each time a particular file system or directory is accessed.

The size of simple- and spanned-volumes can be dynamically increased during operation.

Logical Disk Manager (LDM)

LDM is the part of the operating system responsible for managing dynamic disks. It manages them as a *disk group*. Each disk in a group contains information about all other disks and the volumes configured on these disks in a replicated transactional database written in an area (1MB - 8MB in size) at the beginning of the disk. In Windows 2000, the volume configuration information is no longer stored in the registry.

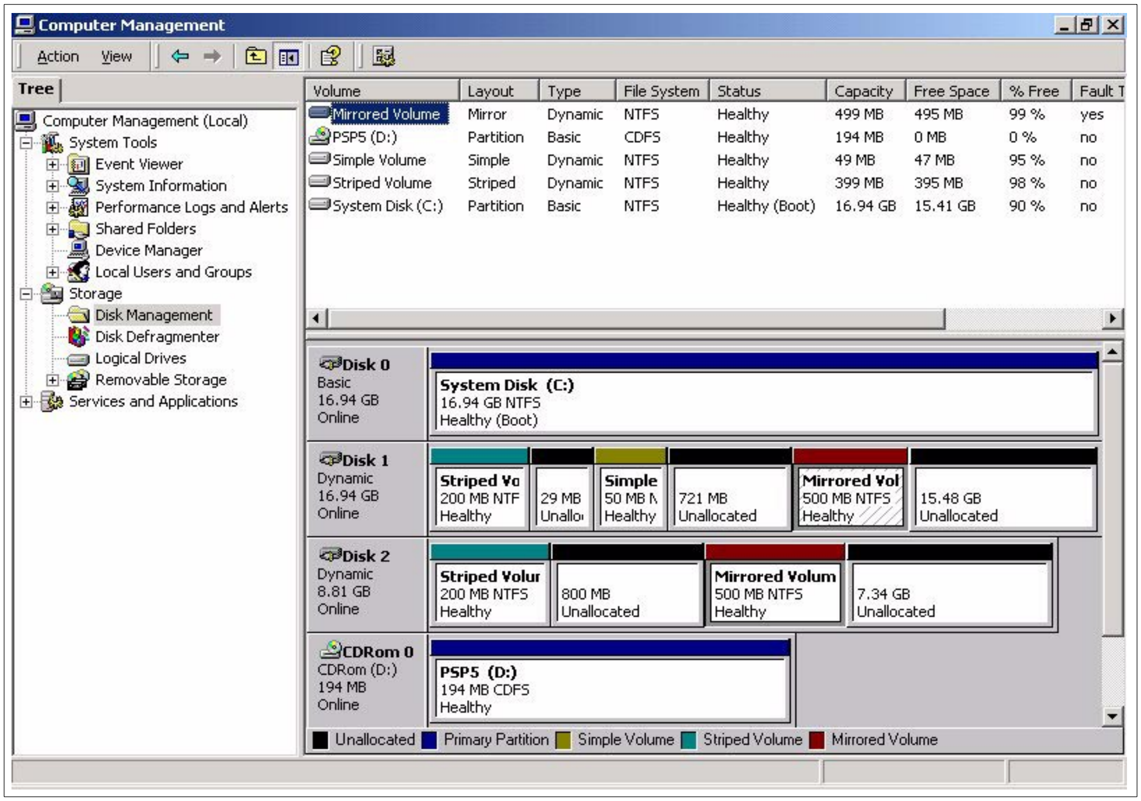
For users who are familiar with Windows NT terminology, Table 8 compares Windows NT and Windows 2000 terminology.

Table 8. Windows NT and Windows 2000 terminology compared

Windows NT	Windows 2000
Partition	Volume
System and Boot Partitions	System and Boot Volumes
Active Partition	Active Volume
Extended Partition	Volumes and Unallocated space
Logical Drive	Simple Volume
Volume Set	Spanned Volume
Stripe Set	Striped Volume
Mirror Set	Mirrored Volume

6.2.1.2 Configuring and managing disks and volumes

An MMC Snap-in called Disk Management is used to create, change, or delete volumes and disks as seen in Figure 44 on page 137.



Volume	Layout	Type	File System	Status	Capacity	Free Space	% Free	Fault T
Mirrored Volume	Mirror	Dynamic	NTFS	Healthy	499 MB	495 MB	99 %	yes
PSP5 (D:)	Partition	Basic	CDFS	Healthy	194 MB	0 MB	0 %	no
Simple Volume	Simple	Dynamic	NTFS	Healthy	49 MB	47 MB	95 %	no
Striped Volume	Striped	Dynamic	NTFS	Healthy	399 MB	395 MB	98 %	no
System Disk (C:)	Partition	Basic	NTFS	Healthy (Boot)	16.94 GB	15.41 GB	90 %	no

Disk	Volumes
Disk 0 Basic 16.94 GB Online	System Disk (C:) 16.94 GB NTFS Healthy (Boot)
Disk 1 Dynamic 16.94 GB Online	Striped Vo 200 MB NTF Healthy
	29 MB Unallo
	Simple 50 MB N Healthy
	721 MB Unallocated
	Mirrored Vol 500 MB NTFS Healthy
	15.48 GB Unallocated
Disk 2 Dynamic 8.81 GB Online	Striped Volur 200 MB NTFS Healthy
	800 MB Unallocated
	Mirrored Volum 500 MB NTFS Healthy
	7.34 GB Unallocated
CDRom 0 CDRom (D:) 194 MB Online	PSP5 (D:) 194 MB CDFS Healthy

■ Unallocated ■ Primary Partition ■ Simple Volume ■ Striped Volume ■ Mirrored Volume

Figure 44. Disk Management utility

6.2.2 Windows 2000 file systems

Windows 2000 supports all the file systems from previous versions of Windows except HPFS, used in OS/2. It also adds support for several new ones.

The following file systems are supported:

- **FAT (or FAT16)** - This file system was part of the Microsoft DOS operating system. In Windows 2000, it is supported on the top of the FtDisk device driver and has the support of long filenames. The maximum volume size for a FAT16 partition is 2 GB.
- **FAT32** - FAT32 was introduced with the Windows 95 OSR2 and Windows 98 operating systems. The reduced cluster size in this file system results in a 20 to 30 percent increase in disk space efficiency compared to FAT16 file systems. The maximum file size and volume size for FAT32 is 32 GB.

- **Compact Disk File System (CDFS)** - This type of file system allows Windows 2000 to read data from CD-ROM devices. It is implemented in accordance with the ISO 9660 standard with additional support for Joliet disks.
- **Universal Disk Format (UDF)** - This is a file system defined by the Optical Storage Technology Association (OSTA) and is compliant with the ISO 13346 standard. This is the next generation of CDFS and supports various optical media such as DVD, WORM, and CD-R. It supports long and Unicode filenames, Access Control Lists, streams, and sparse files.
- **NT File System (NTFS)** - NTFS is the native Windows 2000 file system. It was designed to provide greater reliability and recoverability of the data and to address several limitations of FAT file systems. The current version is NTFS Version 5. It has several enhancements and new functionalities over NTFS Version 4. The NTFS Version 4 file systems are upgraded to NTFS Version 5 at the time of the first mount of the volumes. The main characteristics of NTFS are:
  - **Reliability** - NTFS logs transaction to its structure (similar to AIX's JFS log), so in case of a failure, utilities like CHDISK can roll-back the activity on the file system and ensure its integrity.
  - **Large disk and large file support** - NTFS allocates clusters of up to 4 KB in size and uses 64-bit addresses to number them; so, 2^{64} x 4 KB is the maximum manageable space, and each file could, at least theoretically, be 2^{64} bytes (16 Exabytes) in size. However, since a spanned volume can use a maximum of 32 disks, the actual maximum size of a file is limited by the size of the hard drives. With 72 GB hard drives, the maximum size of a volume set and a single file is 32 x 72.6 = 2.3 TB. However, using additional hardware can increase this further.
  - **Multiple data streams** - NTFS has the structure of a relational database. All information associated with a file, such as the file name, owner, permissions, and the file's data blocks themselves, are implemented as file attributes. Each attribute is a stream, a simple sequence of bytes. Because file data is just another attribute, new attributes can be added to make the file system flexible. You will find a more detailed description of NTFS in section "NTFS organization" on page 139.
  - **Encryption** - If implemented on NTFS, the Encrypting File System (EFS) enables the user to transparently use the file, although it is encrypted on the disk and cannot be read by any low-level disk examination utility.

- **POSIX compliance** - The NTFS file naming convention is POSIX.1-compliant and supports case-sensitive naming, additional time stamp, and hard links.
- **Sparse files** - Sparse files are very large files with a lot of 'holes' in them (such as large matrixes) that are written in such a way as to occupy minimum disk space.
- **Remote storage** - In Windows 2000, remote media, such as tapes or optical drives, can be transparently integrated into NTFS.
- **Disk quotas** - Disk quotas enable the system administrator to monitor and control the disk space used by individual users.

6.2.2.1 NTFS organization

The core of the NTFS structure is the file called the Master File Table (MFT). An entry for each file on an NTFS volume is stored in a record (or a row) of the MFT and consists of all the file's attributes, such as file size, time and date stamps, permissions, and so on. In order to keep the MFT as contiguous as possible, an area of the volume is reserved for the MFT when NTFS is created. The schematic structure of the MFT is shown in Figure 45.

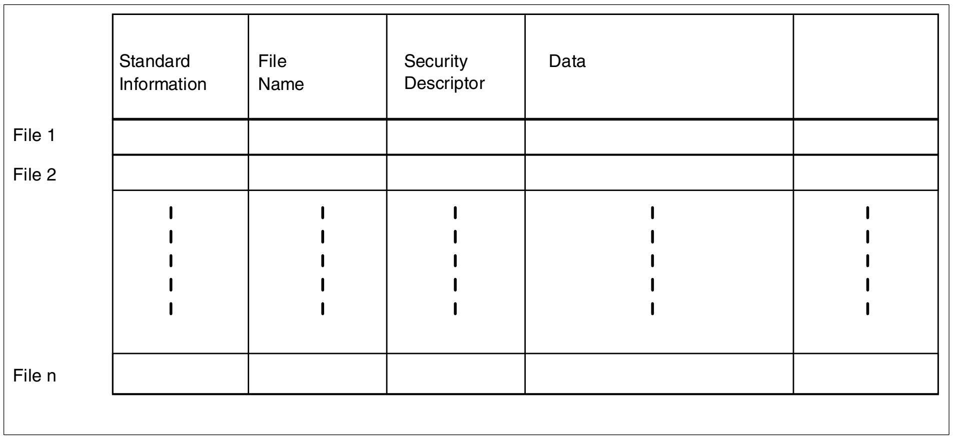


Figure 45. Master File Table structure

The basic attributes of a file are Standard Information, File Name, Security Descriptor, and Data. Each attribute can grow independently of the others. Writing to a file using NTFS terminology means writing bytes into the file Data attribute. Changing a file name means writing into its File Name attribute. When the value of an attribute is directly stored in the MFT, the attribute is called a resident attribute. File attributes that are smaller than the cluster size will be resident. This means that small files will have their data attributes stored in the MFT itself.

For bigger files, NTFS will allocate additional clusters called runs (or extents) on disk to store the data attributes. These attributes are called nonresident attributes because they are not stored in the MFT.

When nonresident attributes are used for large files, NTFS keeps track of the location of these attributes. NTFS uses Virtual Cluster Numbers (VCNs) to number clusters belonging to a file and clusters within a volume. Thus, to retrieve the data blocks (clusters) that are not stored in the MFT, NTFS maps the VCN to Logical Cluster Number (LCN) and stores that information in the data file attribute in the MFT. Figure 46 depicts how VCN-to-LCN mapping works for nonresident attributes used by large files.

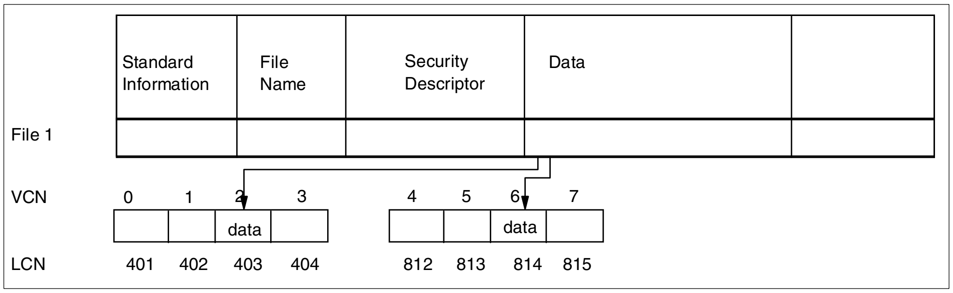


Figure 46. VCN to LCN mapping for large files

NTFS handles directories as well as files (see Figure 47). A directory is actually a file whose data attributes are pointers (or indexes) to files that belong to that directory. If the directory is large, that is, if it contains a very large number of files, NTFS will also use nonresident attributes to store the index of files.

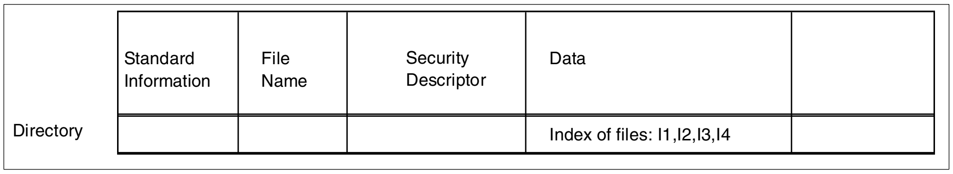


Figure 47. Directory record structure in MFT

6.2.2.2 NTFS cluster

A cluster is the adjustable unit of disk allocation for NTFS and is similar to the AIX term ‘fragment’. A cluster is the smallest amount of disk space that can be allocated to contain a file. The cluster size is defined when a volume is formatted to NTFS. NTFS uses a cluster size of 512 bytes on small disks and a maximum of 4 KB on large disks. The cluster size or cluster factor has a default value to minimize disk fragmentation. That default value varies with the size of the physical volume and is a power of two of the physical sector (one sector, two sectors, four sectors, eight sectors, and so on). It is also manually assignable in a range of 512 bytes to 4 KB. The default cluster size doubles every 512 MB as shown in Table 9.

Table 9. Relation between default cluster size and disk size

Cluster size in bytes	Sectors	Disk drive size
512	1	1 - 511MB
1024	2	512MB - 999MB
2048	4	1GB - 2GB
4096	8	> 2GB

A volume (logical drive) is physically divided into clusters that can be seen as logical blocks or fragments. NTFS allocates clusters without being aware of the physical sectors.

NTFS refers to physical locations inside the hard disk by means of a Logical Cluster Number (LCN). The LCNs enumerate the clusters inside a volume from 0 to n. If a cluster has an LCN equal to 0, this means that this is the first cluster in a volume. To convert an LCN to a physical disk address, NTFS multiplies the LCN by the cluster factor to obtain the physical byte offset of the volume.

The clusters are logically organized in files. NTFS keeps track of single file clusters by means of Virtual Cluster Numbers (VCN) that enumerates the clusters belonging to a particular file from 0 to m.

NTFS relies on a device driver called FtDisk to provide volume management features and for Basic disks and Logical Disk Manager to provide volume management features and fault tolerant features, such as mirroring or RAID-5 for dynamic disks.

6.2.2.3 NTFS volume structure

A volume formatted with NTFS is organized as shown in Figure 48 on page 142.

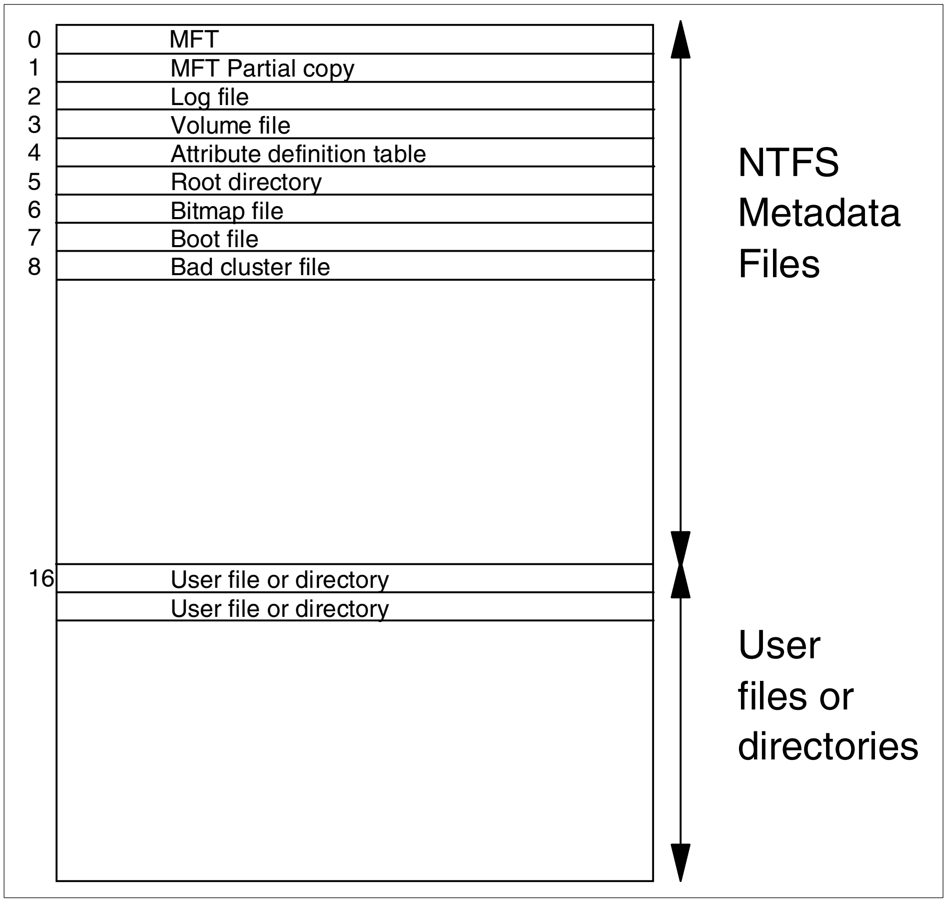


Figure 48. NTFS volume structure

The MFT contains one row for each file or directory. However, there are some specific files called metadata files that are also part of the NTFS volume.

The first file on the volume is actually the MFT itself; the second file is a partial copy of the MFT that contains the first 16 rows of the MFT, namely the metadata files. This is used in case one of the metadata files is corrupted and cannot be accessed.

The third file is the log file. The log file is used to log all transactions issued on the NTFS file system. It is used to recover a file system after an abnormal shutdown.

The volume file contains the volume name, the NTFS version, and integrity information.

The root directory file contains an index of the files and directories stored in the root of the NTFS directory structure.

An important file in the NTFS volume structure is the boot file. This boot file contains the Windows 2000 bootstrap code. This code has to be at a specific disk address. When NTFS accesses a volume, it must mount the volume by reading the physical address of the MFT table in the boot file.

Then, NTFS first looks for the first MFT entry. Reading the Data attribute, NTFS can retrieve the VCN-to-LCN list and store it in memory. NTFS then retrieves the metadata file called Log to record all operations. Finally, it retrieves the Root Directory file and is ready to work with that volume.

Another important metadata file is the Bitmap file that stores the allocation state of a volume.

Last but not least, the Bad Cluster file contains a map of unavailable clusters in a volume.

6.2.2.4 NTFS recoverability

NTFS implements a model based on transaction processing techniques to assure recoverability. It uses a log file saved as a metadata file and a log service called LFS (Log File Service). NTFS works in a transactional manner: Every change on a file or directory is seen as a transaction, and the changes of the file system are logged. After a transaction is successfully completed, a confirmation will be received. This approach ensures that a volume will never be lost because of a crashed file system.

Note

Only file system metadata is fully recoverable; user data is not guaranteed against volume failures. This is similar to the AIX JFS file system.

Several techniques have been implemented to assure that a single transaction that has been committed will appear on the volume even if the operating system fails immediately thereafter. So if a transaction does not reach the committed state, every operation will be rolled back to guarantee file system consistency.

NTFS solves this by implementing a Lazy-Write algorithm. This algorithm allows the file system to raise its performance with timely delayed write-back processes. This means that an application could already have received a confirmation that a file is successfully written without the actual write to disk operation having taken place. The time delay has another effect: When a file cannot be written, there must be an emergency procedure to avoid a data loss. Also, there must be a way, in case of a power loss, to re-create the volume and the file structure and recover the data.

The log file holds all information to redo the transactions when it fails because of a writing error.

Note

The logfile is part of the volume structure. It is not a separate volume so it cannot be created on a different physical disk in order to increase the file system performance during write access.

After every reboot, NTFS performs a disk recovery using a transaction table and a dirty page. The transaction table maintains all operations already started but not yet committed, and the dirty page keeps track of all file system modifications not written to disk. With these tables, NTFS is able to rebuild a possibly corrupt volume.

There are some corruptions that NTFS is not able to handle; an external program called CHKDSK is provided for this purpose. It runs in a command-prompt session of Windows 2000 and can restore broken chains or disagreements between the file table and physical disk sectors by dumping lost clusters into files in the root directory.

6.2.2.5 Bad-cluster remapping

Usually, if a program tries to read data from a bad disk sector, the read operation fails and the data in the allocated cluster becomes inaccessible.

However, in the Windows 2000 environment, if a volume on a Dynamic disk disk is formatted as a fault-tolerant NTFS volume, such as RAID-1 (mirrored volume) or RAID-5 (distributed parity volume), the Windows 2000 Logical Disk Manager driver dynamically retrieves a good copy of the data that was stored on the bad sector and then sends NTFS a warning that the sector is bad. NTFS allocates a new cluster, replacing the cluster in which the bad sector resides, and copies the data to the new cluster. It flags the bad cluster and no longer uses it.This operation is transparent to the user.

6.2.2.6 Data compression

This facility provides compression of regular files and directories. The compression is implemented on a compression-unit basis (16 clusters long), which means that when a file has to be written, it will be divided into blocks of 16 clusters called *compression units*. The compression unit is then compressed, and NTFS determines the new length (measured in clusters) of the compression unit. If the length is less than 16 clusters, indicating some space has been saved, the compression unit will be written to disk; otherwise, the data will be written uncompressed. For example, in Figure 49, the file that was 43 clusters long has been compressed into 25 clusters.

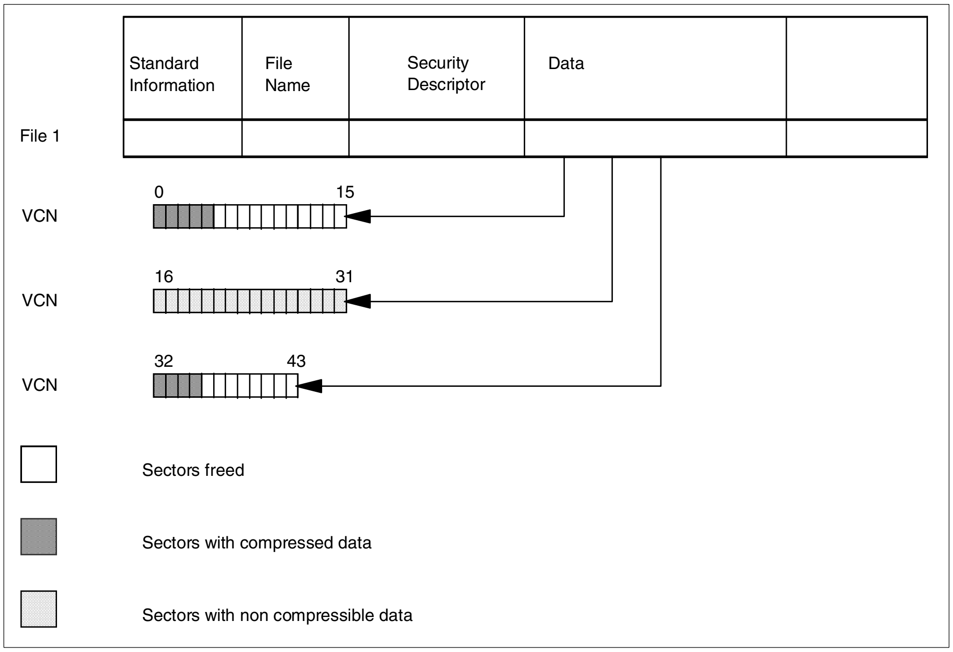
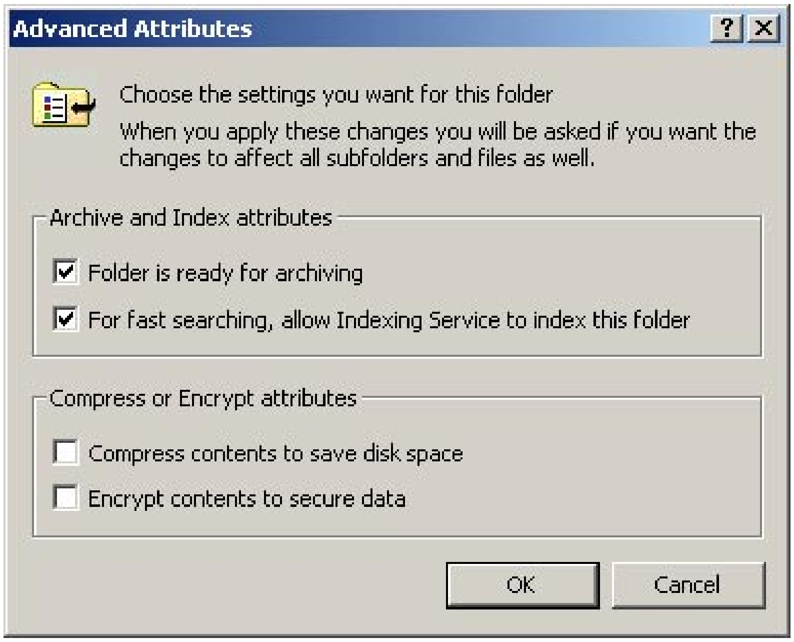


Figure 49. Compressed file structure

Compression is enabled when a volume is formatted, or by right clicking on a file or folder, selecting Properties and then pressing the Advanced button. You should then get a window that resembles Figure 50 on page 146. A spanned volume cannot have compression enabled.



Choose the settings you want for this folder

When you apply these changes you will be asked if you want the changes to affect all subfolders and files as well.

Archive and Index attributes

☑ Folder is ready for archiving

☑ For fast searching, allow Indexing Service to index this folder

Compress or Encrypt attributes

□ Compress contents to save disk space

□ Encrypt contents to secure data

Figure 50. Advanced attributes allow file encryption

6.2.2.7 Encryption

In the Windows 2000 EFS file system, you can encrypt your files and directories. Once encrypted, you can use them in the same way as un-encrypted files or directories; you do not have to manually decrypt them before use. If another user tries to access your file or directory, they would receive an *Access denied* message.

There are certain considerations when you are using encryption:

- Only files and volumes on an NTFS file system can be encrypted.
- You cannot encrypt compressed files or directories.
- You cannot share encrypted files.
- Files tagged for encryption become decrypted if you move them to a non-NTFS file system.
- System files cannot be encrypted.

6.2.2.8 Long file name support

NTFS supports long UNICODE file names, allowing users to save files using, for instance Chinese or Hebrew characters. At the same time, NTFS maintains an 8.3 name for the file so that it can be used by DOS programs or old 16-Bit Windows programs within a Windows 2000 session. NTFS also supports case-sensitive file access for UNIX programs and case-insensitive file access for DOS, OS/2, and Windows programs (see Figure 51).

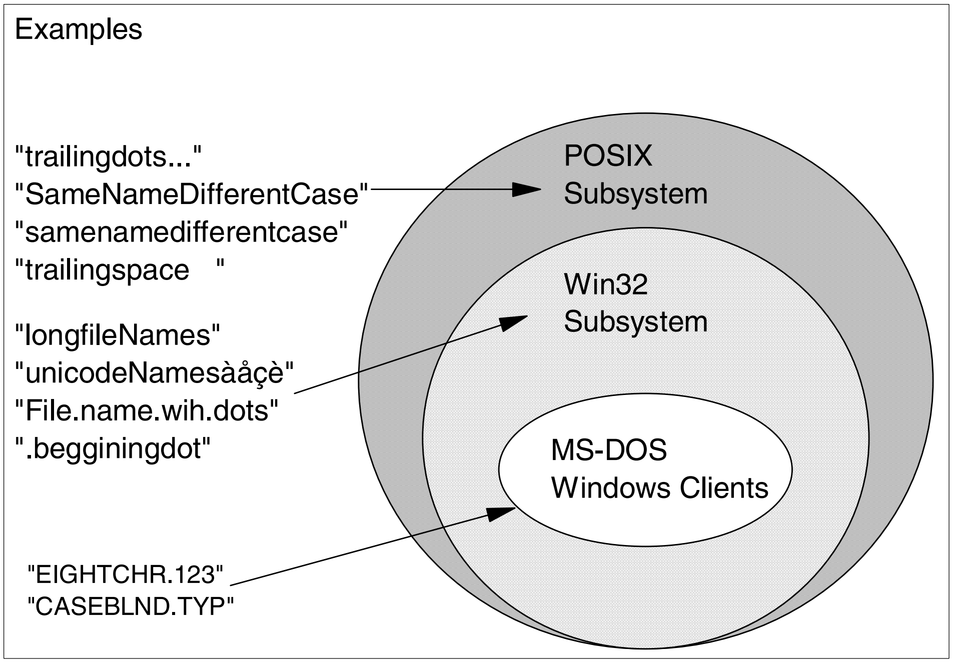


Figure 51. Windows 2000 file namespaces

The MS-DOS file names are fully-functional aliases for the NTFS files and are stored in the same directory as the long file names. The Master File Table (MFT) record for a file with its auto-generated MS-DOS file name is shown in Figure 52.

	Standard Information	File Name	Security Descriptor	Data	MS-DOS File Name
File 1					

Figure 52. MS-DOS file name attribute

The NTFS name and the generated MS-DOS name are stored in the same record and, therefore, refer to the same file. The MS-DOS name can be used to open, read from, write to, or copy the file. If a user renames the file using either the long name or the short file name, the new name replaces both existing names. If the name is not a valid MS-DOS name, NTFS automatically generates another MS-DOS name for the file.

The POSIX subsystem is supported and requires the biggest namespace of all the application execution environments that Windows 2000 supports. The POSIX subsystem can create names that are not visible to the Win32 and MS-DOS applications including names with trailing periods and trailing spaces. POSIX hard links are supported while symbolic links are not.

6.2.2.9 Sparse files

A sparse file is a file with regions of unallocated data in it. For example, a large matrix has only one column and one row with non-zero values, all the others are equal to zero. Such a matrix (if it is large) can allocate a large amount of disk space, but the majority of its contents would only be zeros.

In Windows 2000, you can have a special *Sparse File Attribute* set for such a file. With this attribute, only the allocated data and range data for zeros will actually allocate space on the disk as shown in Figure 53.

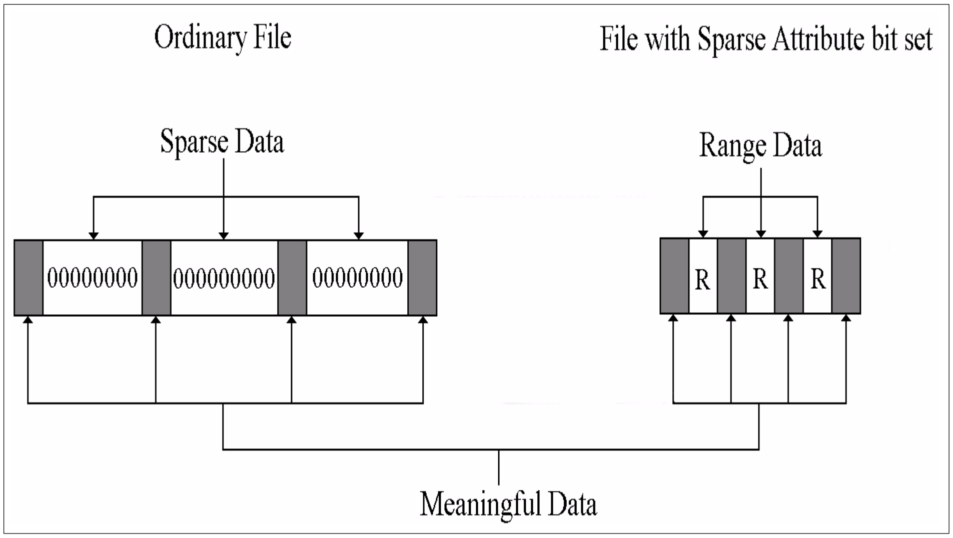


Figure 53. Sparse file structure

6.2.2.10 Removable and remote storage

In Windows 2000, removable media, such as tapes or optical disks, are managed by set of APIs known as Removable Storage Management (RSM). RSM is installed by default and manages all removable media, such as

CD-ROM, DVD, MO, Jaz, Zip, tapes, changers, and juke-boxes. Even if considered removable, floppy drives are not handled by RSM. Different clients use RSM and do not interact directly with particular device drivers; so, they have an *abstract* view of the storage device.

Remote Storage is a service that enables Hierarchical Storage Management (HSM), similar to ADSM HSM. It is based on RSM and on *reparse points* of NTFS, enabling automatic migration of less-used files from disk volumes to tapes or optical disks. This is transparent to the user and he or she still sees the file as if it is located on the local disk. When the file is accessed by a user, it is automatically migrated back to local disk.

6.2.2.11 Distributed File System (DFS)

DFS enables the system administrator to configure the storage (volumes, directories, files) that resides on many systems so that it is accessible by users as if it was resident on only one system. DFS has functionality similar to NFS. The basic features of DFS are:

- **Easy access to files** - Users only need to go to one location on the network (the server), to access all the files as if they were local to the server, regardless of its actual physical location on many servers.
- **Availability** - If the system administrator configures domain-based DFS, all DFS information is stored in the active directory and, therefore, automatically distributed to all domain controllers in the domain, allowing users to access information about DFS from any domain controller in that domain.
  The system administrator also has the ability to replicate the DFS shared folders on other servers in the domain so that the data is available to users even if one server is down.
- **Load balancing** - If the files reside on multiple servers (replication), the DFS service will balance the load on the network and servers by allowing different users to access the files on different servers, although, for the users, the file will appear to be located on just one server.

6.2.3 Windows 2000 RAID support

The following sections describe the various levels of RAID supported by Windows 2000.

RAID 0

RAID 0 is also known as data striping. Conventionally, a file is written to (or read from) a disk in blocks of data. With striping, the information is split into chunks (a fixed amount of data), and the chunks are written to (or read from) a series of disks in parallel.

RAID 0 is well suited to program libraries requiring rapid loading of large tables or, more generally, for applications requiring fast access to read-only data or for fast writing. RAID 0 is only designed to increase performance; there is no redundancy; so, any disk failures will require a restore from backup.

Windows 2000 supports RAID 0 with striped volumes created on dynamic disks.

RAID 1

RAID 1 is also known as disk mirroring since duplicate copies of each chunk of data are kept on separate disks, or, more often, each disk has a twin that contains an exact replica (or mirror image) of the information. If any disk in the array fails, the mirrored twin takes over. Read performance can be enhanced because the disk with its actuator closest to the required data is always used, thereby minimizing seek times. RAID 1 is best suited for applications that require high data availability, good read response times, and where cost is a secondary issue since 50% of the total disk size is required for mirrored data.

Windows 2000 supports RAID 1 with mirrored volumes created on dynamic disks.

RAID 5

RAID 5 is one of the most capable and efficient ways of building redundancy into the disk subsystem. The way redundancy is implemented, capacity loss is equal to one of the drives in the array and data striping provides the read performance gains from RAID 0 and RAID 1. The principles behind RAID 5 are very simple and are closely related to the parity methods sometimes used for computer memory subsystems. In memory, the parity bit is formed by evaluating the number of 1 bits in a single byte. For RAID 5, if we take the example of a four-drive array, three stripes of data are written to three of the drives and the bit-by-bit parity of the three stripes is written to the fourth drive. As an example, we can look at the first byte of each stripe and see what this means for the parity stripe. Let us assume that the first byte of stripes 1, 2, and 3 are the letters A, B, and G respectively. The binary code for these characters is 01000001, 01000010 and 01000111 respectively.

We can now calculate the first byte of the parity block. Using the convention that an odd number of 1s in the data generates a 1 in the parity, the first parity byte is 01000100 (see Table 10 on page 151). This is called Even

Parity because there is always an even number of 1s if we look at the data and the parity together. Odd parity could have been chosen; the choice is of no importance as long as it is consistent.

Table 10. Generation of parity data for RAID 5

Disk1 “A”	Disk 2 “B”	Disk 3 “G”	Disk 4 Parity
0	0	0	0
1	1	1	1
0	0	0	0
0	0	0	0
0	0	0	0
0	0	1	1
0	1	1	0
1	0	1	0

Calculating the parity for the second byte is performed using the same method, and so on. In this way, the entire parity stripe for the first three data stripes can be calculated and stored on the fourth disk. The presence of parity information allows any disk to fail without loss of data.

In the above example, if drive 2 fails (with B as its first byte) there is enough information in the parity byte and the data on the remaining drives to reconstruct the missing data. Because of this, a RAID 5 array with a failed drive can continue to provide the system with all the data from the failed drive.

Performance will suffer, of course, because the OS has to look at the data from all drives when a request is made to the failed one. However, this is still better than losing the system completely. A RAID 5 array with a failed drive is said to be critical as the loss of another drive will cause lost data.

The simplest implementation would always store the parity on disk four (in fact, this is the case in RAID 4, which is hardly ever implemented for the reason about to be explained). Disk reads are then serviced in much the same way as a RAID 0 array with three disks. However, writing to a RAID 5 array would then suffer from a performance bottleneck. Each write requires that both real data and parity data are updated. Therefore, the single parity disk would have to be written to every time any of the other disks were modified. To avoid this, the parity data is also striped, spreading the load across the entire array.

Windows 2000 supports RAID 5 with dynamic disks.

6.2.4 Security and the user model

Windows 2000 and NTFS data security are based on several layers. The first layer has two parts called *User Authentication* and *Access Control*, and both are based on the use of the active directory (see Chapter 7, “Security” on page 153, for more details). The second layer of security is data encryption with Encrypting File Systems (EFS) and Digital Signatures, which signs and protects software components.

Windows 2000 Server also supports a Macintosh security model that simulates an Apple File Server. UNIX applications will see security that conforms to the POSIX model.

Chapter 7. Security

System security is becoming more and more important, especially in a network environment where resources are shared among computers and users. While it is relatively easy to ensure a high level of security on stand-alone systems, it is not so easy to guarantee the same level of security when dealing with a large number of systems on a network.

System security is as much an art as it is a philosophy. The philosophy of system security is based on a realistic understanding of what one needs to be protected from whom and how much effort or resources to expend to accomplish the task.

The following are considerations when securing a system:

- Threat assessment, determining the likelihood of being attacked maliciously compared to accidental damage assessment
- Detection of security-related incidents
- Available tools (those native to the operating system as well as third-party tools)
- The amount of time to devote to security
- Knowledge about security practices including configuration, design, and monitoring.
- The proper role of system backups for disaster recovery

With systems spread across departments, buildings, and even miles, managing security consistently becomes more difficult. Besides being distributed, systems today are connected to other systems via LANs and WANs.

While both AIX and Windows 2000 employ many levels of security to protect the systems and the network, no amount of protection on the operating system or network can guard against employees or insiders misusing the operational trust that has been given to them to function in their job. Both operating systems, however, can provide a high level of protection from outsiders.

7.1 Security standards

Requirements for secure systems are defined by the U.S Department of Defense National Computer Security Center (NCSC) in a publication called

Trusted Computer System Evaluation Criteria (TCSEC), also known as the Orange Book. A network version of these security criteria, called the Trusted Network Interpretation, has been developed and is known as the Red Book.

Fundamentally, the Orange Book defines four divisions (A, B, C, and D) of secure components. Each division is associated with a level of security and is divided into one or more classes. For a system to adhere to a level of security, it must meet all the criteria defined in that level. These criteria concern four domains: Security, Policy, Accountability, and Assurance and Documentation.

These are the different divisions and their classes as defined in the Orange Book:

- Division D - No protection
  - Class D1 - No security
- Division C - Discretionary protection
  - Class C1 - Discretionary protection
  - Class C2 - Controlled access
- Division B - Mandatory protection
  - Class B1 - Labeled protection
  - Class B2 - Structured protection
  - Class B3 - Security domain
- Division A - Verified protection
  - Class A1 - Verified design

The C2 certification level is characterized by:

- **Discretionary Access Controls (DAC)** - These access controls must be able to grant or deny access to the resource at the user level (user granularity). Defined groups must specify individuals. Only authorized users can give a user access to a resource that has no access permissions.
- **Object Reuse Protection** - The operating system must protect data stored in memory for one process so that it cannot be used by another process. Another important feature is that, when a file is deleted, it can no longer be accessed by another user or process.
- **Identification and Authentication** - Each user must be identified uniquely. The system should be able to track all user activity by associating the user's unique ID with actions taken by that user.

- **Audit** - System administrators must be able to audit security-related events and the actions of individual users. Access to audit data must be limited to authorized administrators.

Figure 54 is from the article *Trusted Products Evaluation*, by Dr. Santosh Chokhani, from the July 1992 Communications of the ACM, and it shows the features of each division:

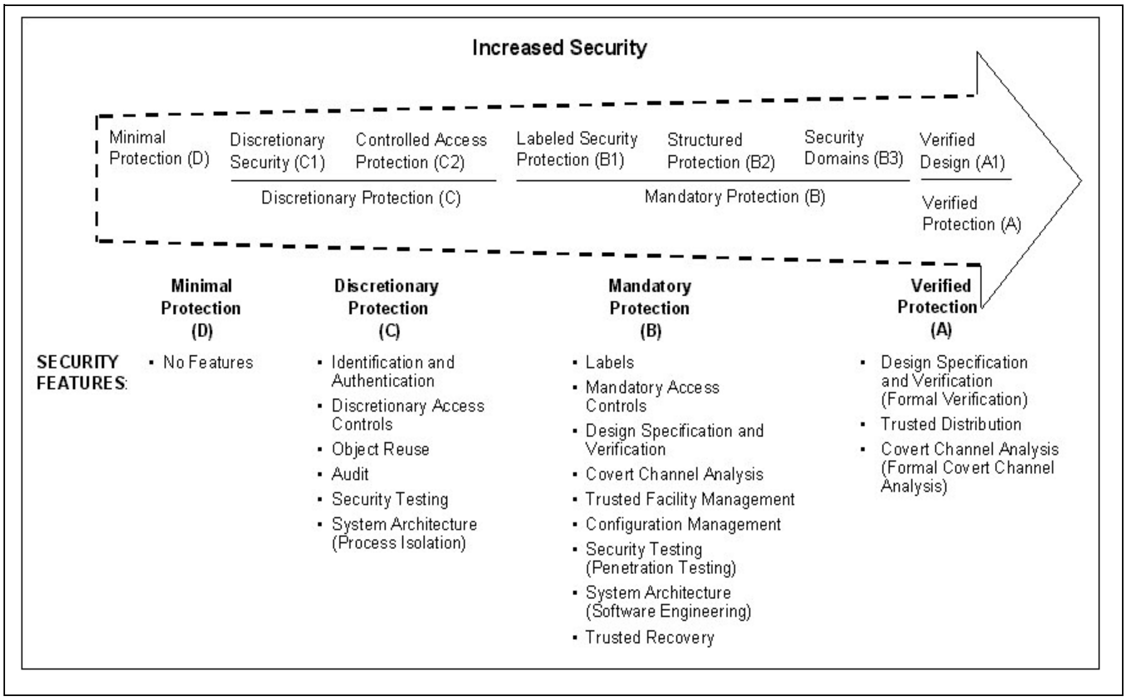

Increased Security
Minimal Protection (D)
Discretionary Security (C1)
Controlled Access Protection (C2)
Labeled Security Protection (B1)
Structured Protection (B2)
Security Domains (B3)
Verified Design (A1)
Discretionary Protection (C)
Mandatory Protection (B)
Verified Protection (A)
Minimal Protection (D)
Discretionary Protection (C)
Mandatory Protection (B)
Verified Protection (A)
SECURITY FEATURES:
• No Features
• Identification and Authentication
• Discretionary Access Controls
• Object Reuse
• Audit
• Security Testing
• System Architecture (Process Isolation)
• Labels
• Mandatory Access Controls
• Design Specification and Verification
• Covert Channel Analysis
• Trusted Facility Management
• Configuration Management
• Security Testing (Penetration Testing)
• System Architecture (Software Engineering)
• Trusted Recovery
• Design Specification and Verification (Formal Verification)
• Trusted Distribution
• Covert Channel Analysis (Formal Covert Channel Analysis)

Figure 54. Trusted computer system evaluation criteria - Rating scale

Note that, outside the U.S., other countries developed their own criteria. For example, France, Germany, the Netherlands, and the UK developed a unified evaluation criteria document called the Initial Technology Security Evaluation Criteria (ITSEC).

A common criteria is under development in order to simplify the evaluation of system security. Its purpose is to merge the U.S. criteria with the Canadian and European criteria to create a unique set of evaluation criteria. It is not the purpose of this book to cover all the different security criteria and flavors.

7.1.1 Other security-related resources

There are many resources available, either through books, service providers, or on the Internet. Here are a few well-known sources from the Internet:

- **Computer Emergency Response Team (CERT)** - Originally developed in 1988 by the Defense Advanced Research Projects Agency or DARPA, CERT houses a repository of security-related bulletins and security-related tools. CERT can be found at:

  `http://www.cert.org/`

- **Forum of Incident Response and Security Teams (FIRST)** - Since 1988, FIRST has brought together over 30 different security incidence response teams from government, academic, and commercial sectors. The Internet page for FIRST can be found at:

  `http://www.first.org/`

Of course, there are many more sources of quality security-related information available. Many countries have their own CERT affiliates or clearing houses. The sources mentioned above are good starting points and will each point the reader to several other sources of information.

7.2 AIX security

This section describes the components that make up the AIX security model.

7.2.1 User management and authorization controls

Hackers will first attempt to penetrate a system by logging in to that system. Before a user can log into an AIX system, the administrator (known as root) must set up an account for that person. It is possible to set up the account so that it expires on a certain date, if necessary. The administrator sets the initial password, and, on the first login, the user must change it.

The support for identification and authentication modules has been enhanced in AIX 5L. User identification and authentication can be handled separately in AIX 5L and can be executed by different operating system modules.

7.2.1.1 User identification

What user IDs exist and what attributes they have are all a part of user identification. Traditionally /etc/passwd contains all the user identification information required by AIX, but using the LDAP or Kerberos 5 modules is now supported as a way of supporting a more robust and thorough user identification method.

Kerberos 5 integration

The AIX 5L operating system allows the system administrator to replace the default login process with Kerberos 5 authentication. Using Kerberos 5, once a user is logged in their ID will acquire all appropriate network and system credentials.

In previous AIX releases, DCE and NIS were supported as alternate authentication mechanisms. AIX Version 4.3.3 added LDAP support and the initial support for specifying a loadable module as an argument for the user/group managing commands, such as `mkuser`, `lsuser`, `rmuser`, and so forth. But this was only generally documented in the /usr/lpp/bos/README file. AIX 5L is now offering a general mechanism to separate the identification and authentication of users and groups, and defines an application programming interface (API) that specifies what function entry points a module has to make available to be able to work as an identification or authentication method. This allows for a more sophisticated and customized login method beyond what is provided by the standard ones based on /etc/passwd or DCE.

Lightweight Directory Access Protocol (LDAP) support

Starting with AIX 4.3.3, LDAP support became integrated within AIX. LDAP is a means of maintaining consistent user data across a client-server environment. With AIX 5L, LDAP can be used for user identification and for secure network host name resolution for Domain Name Services (DNS).

IBM SecureWay Directory Version 3.2

Upgraded from Version 3.1.1 which was available in AIX 4.3.3, AIX 5L introduces IBM Secureway Directory Service Version 3.2 for user identification. Once installed and configured, user data is securely stored, replicated, and managed by the SecureWay Directory server.

Secureway Directory 3.2 includes LDAP Version 3, which supports the IETF protocol outlined in RFC 2251, schema, RootDSE, UTF-8, referrals, and a Simple Authentication and Security Layer (SASL) for user authentication.

Network Information Service (NIS) support

NIS is a another network service like Kerberos or LDAP that keeps and controls data of users and their attributes. In large distributed computing environments NIS can be used to keep a consistent record of all users within the network topology.

Network Information Service Plus (NIS+) support

NIS was replaced by NIS+ and was ported to AIX Version 4.3.3. This new version of NIS allows not only for the identification and directory services of users but also for user authentication and password verification.

NIS+ is cover in more detail in Section 7.2.2.4, “NIS+” on page 171.

7.2.1.2 User authentication

Verifying a user through the means of a certificate or password is user authentication. When logging into an AIX system you are first prompted for a user ID, then the system prompts for a password to authenticate the user is who they say they are. As in previous versions of AIX, AIX 5L allows for user authentication to be handled separately from user identification. This means a system administrator could set up Kerberos 5 for user identification and DCE for user authentication. In this scenario, the Kerberos module would be responsible for keeping a record of all the information on the user ID entered, and DCE would be responsible for verifying the password entered is the correct one for the user ID trying to be used.

AIX can also be configured to force users to select passwords with certain characteristics. By doing this, it reduces the likelihood of a hacker guessing a password or using a dictionary search to match the password and being able to access the users account. This can be configured on an individual basis. The following rules can be set by the administrator:

- Minimum number of alpha characters
- Minimum number of non-alpha characters
- Maximum repeated characters
- Minimum time between changes
- Words not in a specified dictionary
- Minimum time or number of password changes before a password can be reused

It is also possible for the administrator to limit when and how individual users can access the system. Users may be restricted to a certain terminal or time of day and day of the week to log in on.

At the time of user creation, a home directory must be specified. This is usually /home/username. All the user’s personal data will be stored in this directory by default. This directory is local to the system on which the user logs in, but it can be remote if accessed through network file systems (NFS) or a distributed computing environment/distributed file system (DCE/DFS). There will be more information on these in later chapters.

A profile is also created in the user's home directory (.profile file) and is executed every time the user logs in. Variables can be set in this file and specific applications started.

Once this has been completed by the administrator, a user will be able to log in to the AIX system.

7.2.1.3 Logon process

A user must be authenticated to the AIX operating system by entering a valid user name and, optionally, a valid password that is never echoed in clear text when the user authenticates. On a secure system, all accounts either have passwords or are invalidated. The default is to have the user ID checked against entries in the /etc/passwd file, and the password is checked against the /etc/security/passwd file on the local system.

Once a login session is granted, the user's environment is created. The system sets up a global environment and then proceeds to set up the user's private environment. The global environment is usually created from the /etc/profile and the /etc/environment files that are system-wide files. These files are used to set environment variables for most users on the system. They can also be used to set read-only environment variables that a user cannot modify or remove.

A user's private environment comes from the user's .profile file and other files usually contained in the user's home directory.

7.2.1.4 Login session tracking

Each time a user successfully logs onto the system or unsuccessfully attempts to log into the system, an entry is logged into one or more files. Some of these files may need to be pruned from time to time to keep their sizes manageable. Often, system administrators will manage these files on a daily or weekly basis by copying them to a safe place to analyze them for abusive login patterns or simply to preserve the information contained in them. Either way, these tasks can be automated with the crontab facility.

These files are:

- **/etc/utmp** - This file lists all users currently logged onto the system. It is a formatted ASCII file and cannot be edited or viewed directly. Reading it requires use of the `/usr/bin/who -a` command. If this file is corrupted or missing, no output is generated from that command.

- **/var/adm/wtmp** - This is another formatted ASCII file that keeps track of users who have successfully logged onto the system historically. The who command is used to examine this file, which contains connect-time accounting records.
- **/etc/security/failedlogin** - This formatted ASCII file keeps track of all failed login attempts. This is the one to watch if you suspect that unauthorized persons are attempting to gain access to the system.
- **/etc/security/lastlog** - This flat ASCII file keeps track of the most recent login status for each user. It is a stanza-based file that contains a stanza for each defined user of the system. Information regarding both the last successful and last unsuccessful logins is kept in this file.
- **/var/adm/sulog** - This is a flat ASCII file that keeps an historical record of all users that have *su-ed* (switched to another user account by using the su command) or attempted to su to another user's account. The su command can be used by any user on the system to become any other user including the root user. However, for the su to be successful, any non-root user will have to enter the password of the user that they want to switch into. A plus sign (+) indicates the su was successful, and a minus sign (-) indicates the attempt was not successful.

7.2.1.5 Administrative roles

As mentioned previously, AIX has a root user that is the administrator of the system. This ID is needed for the installation and setup of AIX and to create additional users. To protect anyone other than the administrator from having to use the root ID and to divide the administrative tasks among 'trusted' users, AIX Version 4.2.1 and above have separated these roles and privileges. Users can now have one or more of the following administrative roles:

- **ManageBasicUsers/ManageAllUsers** - This role allows any user to act as root, add and remove users, change a users information, modify audit classes, manage groups, and change passwords.
- **ManageBasicPasswds/ManageAllPasswds** - This role allows a user to alter the passwords of other users.
- **ManageRoles** - This role allows a user to create, change, remove, and list roles.
- **ManageBackupRestore** - This role allows a user to archive and restore file systems and directories.
- **ManageBackup** - This role allows a user to archive file systems and directories.

- **ManageShutdown** - This role allows a user to shut down, reboot, and halt a system.
- **RunDiagnostics** - This role allows a user to run diagnostics on a system.

Pre-defined users

AIX has some predefined users. These users are already members of the predefined groups that come standard with AIX. The predefined groups are explained later in this chapter. The predefined users are:

- **root** - This is the owner of all resources.

  The system administrator, commonly referred to as root or the super user, is the master user of the system. This is the person who has the responsibility of securing the system. For the most part, the root user is not screened from issuing any commands or operations. When hackers try to access a UNIX system, their aim is to become the root user so that their actions remain hidden. This is why securing the system is so important.

  One of the first access/permission related checks is to see if the user issuing a command or request is the root user. If they are, no further permission checking is performed. The command is performed no matter how adverse it may be to the overall health and safety of the system, although some commands do have the occasional verification steps included. For the most part, it is assumed that the root user knows what he or she is doing. For this reason, it is often wise to never log in as the root user; rather, log in as a user and switch to root using the `su` command. This leaves a log of what is being done in the /etc/security/sulog file.
- **daemon** - This owns the system daemons, such as the printer and scheduling daemons (printq and cron).
- **bin** - This owns the system executable files.
- **sys** - This owns the system devices.
- **adm** - This owns the system utilities.
- **imnadm** - This owns the documentation subsystem.
- **netinst** - This owns the network installation utilities.
- **uucp** - This owns the uucp program.
- **nuucp** - This is the default account for `uucp` connections.
- **guest** - This is the guest account.
- **nobody** - This is used by, for example, NFS when a root user tries to access a file mounted via NFS. It has the same rights as others.
- **lpd** - This is the owner of the printer spool utilities.

7.2.1.6 Access Control Lists (ACLs)

Once a user has been authenticated, authorization must be granted to a single object resource (file or directory) by the subjects (users or processes). AIX provides standard UNIX permissions for owner, group, and others that are read (r), write (w), and execute (x). This is mentioned in more detail in the next section.

In addition, AIX also allows for the use of a B3 security feature (the Access Control Lists or ACLs) to further define access permissions to files and directories. In general, these ACLs are used to specify individual users in a more granular way. ACLs allow a user to define specific permissions to a list of discrete users, beyond that provided by the regular User-Group-Others permissions methodology.

The commands to deal with ACLs are included in the AIX operating system. These commands are `aclget`, to get the ACL of a file, `aclput` to set the ACL, and `acledit` to get and set the ACL. However, the use of ACLs is strictly optional and is not global in nature. That is to say, one directory can use ACLs while another may use the standard Owner-Group-Other permissions. By default, the base AIX operating system does not use ACLs to control access to system files.

7.2.1.7 File and directory permissions

Whether or not a user can access a file or directory is controlled by the file permissions. Every file and directory has an owner. For new files/directories, the user who creates the file/directory is the owner. The owner assigns an access mode to the file/directory. Access modes grant other system users permission to read, modify, or execute the file or directory. Only the owner or users with root authority can change the access mode of a file or directory.

There are three classes of users: User/owner, group, and all others. Access is granted to these groups in some combination of three modes: Read, write, or execute. When a new file is created, the default permissions are read, write, and execute for the user who created the file. The other two groups have read and execute permission. Table 11 illustrates the default file access modes for the three sets of user groups:

Table 11. Default file access modes

	Read	Write	Execute
Owner	Yes	Yes	Yes
Group	Yes	No	Yes
Others	Yes	No	Yes

Files can be read (r), written (w), or executed (x). The system determines who has permission and the level of permission they have for each of these activities. Access modes are represented in two ways in the operating system: Symbolically and numerically.

Access modes are represented symbolically as follows:

- **r** - Indicates read permission, which allows users to view the contents of a file.
- **w** - Indicates write permission, which allows users to modify the contents of a file.
- **x** - Indicates execute permission. For executable files (ordinary files that contain programs), execute permission means that the program can be run. For directories, execute permission means that the contents of the directory can be searched.

For example, a file with the access modes set to rwxr-xr-x gives read and execute permission to all three groups, but write permission only to the owner of the file. This is the symbolic representation of the default setting.

Numerically, read access is represented by a value of 4, write permission is represented by a value of 2, and execute permission is represented by a value of 1. The total value between 1 and 7 represents the access mode for each group (user, group, and other). Table 12 illustrates how to determine the numerical values for each level of access:

Table 12. Octal file access representation

Total Value	Read	Write	Execute
0	-	-	-
1	-	-	1
2	-	2	-
3	-	2	1
4	4	-	-
5	4	-	1
6	4	2	-
7	4	2	1

When a file is created, the default file access mode is 755. This means that the user has read, write, and execute permission (4+2+1=7), the group has read and execute permission (4+1=5), and all others have read and execute permission (4+1=5).

There are also special permissions that can be set for even greater flexibility. They are the SUID (set UID), the SGID (set GID), and the tacky bit.

- **Set UID (or suid) bit: s** - This set-on-execution-bit causes the program to run under the user ID of the program owner rather than the user ID of the user running the program.
- **Set GID bit (or sgid) bit: s** - The file owner and root user have the authority to change the permissions for any/all file system objects of which he or she is the designated owner. This bit is used to force ownership of newly-created files to the current owner of the directory in which the file resides.
- **Tacky bit (t bit)** - When this bit is set on a directory, only the file owner can link or unlink a file in that directory. In the case of a file, the t-bit sets the save-text attribute (the file is not unmapped after usage and is kept in memory). When used as the save-text attribute, this bit is known as the sticky bit.

> **Note**
>
> Creating a file is analogous to linking, and deleting a file is similar to unlinking, from a UNIX perspective.

The basic search order used to allow or deny access to a file is:

1. User
2. Group
3. Others

If a user is granted read access by group permissions, the permissions for *others* are not checked.

7.2.1.8 Groups

Each user on the system is a member of at least one group. These groups are comprised of users that require similar access rights and permissions. By using the group permissions, access to a file or directory can be granted to a group of users so that many people can access that file with ease. The default group used for most users is called the staff group. System administrators can create groups and assign users to groups as needed. An individual user can be a member of several groups simultaneously.

The following groups are predefined in AIX:

- **system** - This group is used by system administrators. The root user is a member of this group by default.
- **staff** - This is the default user group. The users daemon and netinst are members of this group by default.
- **bin** - This is a system group. The users root and bin are members of this group by default.
- **sys**- This is a system group. The users root, bin, and sys are members of this group by default.
- **adm** - This is a system group. The users bin and adm are members of this group by default.
- **uucp** - This group contains the uucp and nuucp users.
- **mail** - Members of this group are able to use the mail commands. There are no default members.
- **security** - This group is used by security users. The root user is a member of this group by default.
- **cron** - This group is used for scheduling activities. The root user is a member of this group by default.
- **printq** - This group is used for printer administration. There are no default members.
- **audit** - This group is used for auditing the system. The root user is a member of this group by default.
- **ecs** - This group is used for IBM connection facilities and has no default users.
- **nobody** - The users, nobody and lpd, are members of this group by default.
- **perf** - No default members.
- **shutdown** - No default members.
- **imnadm** - The user, imnadm, is a member of this group by default.

7.2.1.9 Security-related files

AIX uses several files to manage and maintain security aspects of the operating system and the user accounts. Most security-related files can only be accessed by the operating system or by the administrative user. Descriptions of the most frequently used files follow.

/etc/passwd

This is the master users list. This file can normally be viewed by any user on the system, but it cannot be modified by non-privileged users. It contains one line for each user of the system. A sample line might look like this:

```
joeuser:!:200:200:Joe the User:/u/joeuser:/bin/ksh
```

Each line is a list of attributes separated by colons:

- The user ID (`joeuser`) is limited to eight characters.
- The password attribute can be an asterisk, which indicates that the password is invalid, or, as in this case, an exclamation point, which indicates that the password is in the /etc/security/passwd file.
- The UID of the user.
- The primary group ID (GID) of which the user is a member. The primary group is the default group for newly-created files and directories.
- The Gecos field is a free-form field that usually contains identifying information, such as the user's full name.
- The home directory for this user is the user's private directory.
- The user's default shell, although the user can normally execute other shells if so desired.

/etc/security/passwd

This is the password file. It is a stanza-based file that contains one stanza per user. A sample entry might look like this:

```
joeuser:
password = DD1cQ1OctBCn.
lastupdate = 818269778
flags = ADMCHG
```

- `password` is the encrypted password for that user.
- `lastupdate` is the last time the password was changed. This is used in cases where password aging has been enabled.
- `flags` are additional flags turned on by the system. In this case, the `ADMCHG` flag is on indicating that the system administrator has set the user's password. The user will be automatically prompted to change it the first time he or she logs onto the system.

Note

The user may be prevented from issuing certain commands on that system until the ADMCHG flag is removed by changing the password.

/etc/security/user

This is the main user attribute file on an AIX system. It is a stanza-based file that contains a stanza for each user. Additionally, it contains a special stanza called default. The default stanza applies to all users of the system, but the user's own stanza entries will override settings from the default stanza.

There are far too many entries to go into any detail about this file. It is well documented in the header of the file and InfoExplorer. It can be easily modified with an editor or through the SMIT interface by the administrative user only.

/etc/security/login.cfg

Although similar in nature to the /etc/security/user file, this file manages or defines the system resources. This file defines which shells are available on the system and things, such as which ports can be used as well as the default banner that will be displayed when a user logs on to the system.

/etc/security/limits

This is another stanza-based file that uses a default stanza. This file controls how much of the system resources a user can consume. These resources are mainly CPU and memory resources as opposed to disk resources. There are other files in the /etc/security directory that control what the default attributes of a new user will be and how they are created. In addition, certain files that track invalid login attempts are stored in the /etc/security directory as well as backup copies of important files, such as /etc/password and so on.

Note

Passwords stored in the /etc/security/passwd file are encrypted.

Access to the /etc/security directory is limited to the administrative user and to the security user only.

7.2.2 Network security

The security policy for networking is an extension of the security policy for the operating system, and it consists of the following major components:

- User authentication

- Connection authentication
- Data import and export security

User authentication is provided at the remote host by a user name and password, the same as when a user logs in to the local system. Trusted TCP/IP commands, such as `ftp`, `rexec`, and `telnet`, have the same requirements and go through the same verification process as trusted commands in the operating system.

Connection authentication is provided to ensure that the remote host has the expected Internet Protocol (IP) address and name. This prevents a remote host from masquerading as another remote host.

Data import and export security permits data at a specified security level to flow to and from network interface adapters at the same security and authority levels. For example, top secret data can flow only between adapters that are set to the top secret security level.

7.2.2.1 TCP/IP

When using TCP/IP, standard TCP/IP security mechanisms can be used. General TCP/IP security rules apply. Some commands in TCP/IP provide a secure environment during operation. These commands are `ftp`, `rexec`, and `telnet`. The `ftp` function provides security during file transfer. The `rexec` command provides a secure environment for executing commands on a foreign host. The `telnet` function provides security for login to a foreign host. These commands provide security during their operation only. That is, they do not set up a secure environment for use with other commands. For securing your system for other operations, use the `securetcpip` command. This command gives you the ability to secure your system by disabling the nontrusted daemons and applications and by giving you the option of securing your IP layer network protocol as well.

AIX provides all the remote commands and mechanisms of other UNIX systems, such as telnet and rlogin for virtual terminal. The only security hole for both commands is that the password flows in clear text over the network. The commands `rcp`, `ftp`, and `tftp` (trivial file transfer), `rsh` and `rexec` (for remote execution), and `rdist` (as distribution utility) refer to various files in the user's home directory and system, such as .rhosts, .netrc, and /etc/host.equiv files. Note that if the system administrator disables the corresponding daemons for these commands in the /etc/inetd.conf file, $HOME/.rhost, /etc/host.equiv, and $HOME/.netrc are automatically disabled.

The system administrator is provided with the capability to disable non-trusted commands and daemons, such as `rcp`, `rlogin`, `rlogind`, `rsh`, `rshd`, `tftp`, and `tftpd` by using the `securetcpip` command. To prevent users from accessing the system with the `ftp` command, the file /etc/ftpuser includes the list of users that do not have access to the system with ftp. Other potentially dangerous daemons are fingerd, rwhod, uucpd, and sendmail. If not used, such daemons should be removed from the system or allowed only locally.

In case an attack is detected, the commands `killall` or `shutdown -m` can be used to kill all the running processes and bring the system into single user or maintenance mode. From there, these commands assess the damage through a review of the log files and audit trail.

For X Window, the only security feature available with the current release of X11/Motif is the `xhost` command that allows users to add or delete host names on the list of machines from which the X server accepts connections.

The `netstat` command allows the system administrator to check the currently-active TCP connections as well as the listening TCP and UPD daemons. `netstat -a` as well as `netstat -f inet` can be used.

The `tcpdump` command allows promiscuous snooping on Ethernet, not only the recording of packets going to and from the machine but also any other packet on the Ethernet. This does not apply to token-ring.

Trusted processes

A trusted program, or trusted process, is a shell script, a daemon, or a program that meets a particular standard of security. These security standards are set and maintained by the U.S. Department of Defense, which also certifies some trusted programs.

Trusted programs are trusted at different levels. Security levels include A1, B1, B2, B3, C1, C2, and D, with level A1 providing the highest level of security. Each security level must meet certain requirements. For example, the C2 level of security incorporates the following standards:

- **Program integrity** ensures that the process will do what it is supposed to do - no more, no less.
- **Modularity** means that the process source code is broken down into modules that cannot be directly affected or accessed by other modules.
- **The principle of least privilege** states that, at all times, a user is operating at the lowest level of privilege authorized. That is, if a user only

has access to view a certain file, the user does not inadvertently also have access to alter that file.

- **The limitation of object reuse** keeps a user from, for example, accidentally stumbling across a section of memory that has been flagged for overwriting but not yet cleared and may contain sensitive material.

TCP/IP contains several trusted daemons and many nontrusted daemons. The trusted daemons have been tested to ensure that they operate within particular security standards.

7.2.2.2 NFS

The Network File System (NFS) is a distributed file system that allows users to access files and directories located on remote computers and treat those files and directories as if they were local. For example, users can use operating system commands to create, remove, read, write, and set file attributes for remote files and directories.

AIX supports the latest NFS protocol update, NFS Version 3. AIX also provides an NFS Version 2 client and server and is, therefore, backward compatible with an existing install base of NFS clients and servers.

NFS provides its services through a client-server relationship. The computers that make their file systems or directories and other resources available for remote access are called servers. The act of making file systems available is called exporting. The computers, or the processes they run, that use a server's resources are considered clients. Once a client mounts a file system that a server exports, the client can access the individual server files (access to exported directories can be restricted to specific clients).

The Network File System (NFS) is shipped with AIX. Standard NFS security rules apply here with the use of the /etc/exports file. This file allows the system administrator to export directories in order to share information over the network. It also specifies which client systems can have access to the exported directories and with which permissions (Read-Only or Read/Write). Also, root access can only be given to root users of specific systems.

In addition, AIX provides ACLs between AIX systems. It also offers the possibility of using Secure RPC for integrity and authentication, but this is restricted outside the U.S. because it contains the Data Encryption Standard (DES).

7.2.2.3 DCE

The Distributed Computing Environment (DCE) is the best standard solution for creating secure client/server applications. DCE provides security built into the RPC mechanism, security services, and tools to be used when creating, using, and maintaining distributed applications.

The security services are:

- **Registry Service** - This security service maintains a database of accounts, principals, groups, and organizations and embodies administrative policies.
- **Authentication Service** - This security service allows two principals/users to authenticate or verify each other’s identities by being considered trusted third parties for the authentication process.
- **Privilege Service** - This security service certifies a principal’s credentials and allows principals to access resources.
- **Login Service** - This security service initializes a user’s DCE security profile and provides a single identity across the distributed environment.
- **Authorization through Access Control List** - This security service compares entries in the file’s or directory’s ACL and determines the requester’s rights to get access to the resource.
- **Secure RPC** - This security service provides authenticated and secure communication between client and server principals.

7.2.2.4 NIS+

Starting with AIX Version 4.3.3, AIX has included a port of the Sun Solaris Version 2.5 NIS+ implementation. This function is provided in addition to the current NIS support, which remains unchanged (and was considered weak). This new naming service provides enhanced capabilities for security management in a distributed system environment.

Security can be managed for a set of systems using a single management point. NIS+ was designed to meet the demanding requirements of networks, which typically range from 100 to 10,000 multi-vendor clients supported by 10 to 100 specialized servers located at various sites.

NIS+ enables system administrators to store information about client addresses, security information, mail information, network interfaces, and network services in central locations where all the clients on a network can access it. Information is incrementally updated and propagated immediately, allowing information to be changed rapidly.

The NIS+ namespace was designed with hierarchical domains to accommodate more distributed networks requiring scalability and decentralized administration. The NIS+ implementation is optimized to support up to 10 replicas per domain. Such a domain may typically have 10,000 table entries. Scalability beyond 1,000 NIS+ clients is best achieved by dividing NIS+ name spaces into different domains to create a hierarchy.

7.2.2.5 Network protocol security

IP Security enhancements starting with AIX Version 4.3.3 include:

- **Improvements to serviceability** - Logic has been added to allow better tracing of the IP Security and Internet Key Exchange (IKE) messages. The output of logging is now formatted in a readable format, and AIX auditing has been implemented. Users can now pinpoint configuration failures and view audit logs to determine security attacks.
- **On-demand tunneling** - Dynamic tunnels only need to be defined one time and will be activated only when traffic matching the criteria set out in the policy was sent or received. This feature is beneficial to users because the functions for negotiating, computing, and refreshing session keys will only be performed when necessary.
- **Web-based System Manager to configure filters** - A GUI-based tool can now be used to configure and manage manual tunnels, static and dynamic filter rules, and the importation and export of tunnel definitions. The tool is now consistent across IP Security and is NLS-enabled; however, SMIT is not available for this option.
- **Filters based on IP address ranges** - IKE Tunnels can be created that specify a range of IP addresses (starting and ending IP address range endpoints) that allow tunneling over multiple IP addresses. It allows users to easily define tunnels for ranges of addresses.
- **Certificate-based use of Digital Signature for IKE Authentication** - IKE tunnels (dynamically negotiated secure tunnels) have been enhanced to use digital certificates for authentication. Authentication is accomplished by signing IKE messages using X.509 certificates. Certificates may be stored locally. This enhancement enables the deployment of Virtual Private Networks (VPNs) with a large number of endpoints. Such network configurations may present a savings in cost and administration of typical leased line installations.
- **AIX SOCKS API -** Allows generic TCP/IP applications to connect to hosts through a generic TCP/IP proxy using SOCKS protocol Version 5. Any application that only makes outgoing TCP connections can take advantage of this API without any code modification because the library

will automatically handle the tunnel creation with a configured SOCKS5 server. Furthermore, the network administrators can configure the API to accept and route workload across multiple SOCKS5 servers. This library enables network administrators to allow limited access to external sites while maintaining network boundary controls.

7.2.3 Trusted computing base (TCB)

TCB can only be installed when AIX is first installed on a system, and after installation TCB allows a system administrator to set parameters for system hardware and software security. TCB will have a record entry for each hardware and software component on the system and will enforce security measures for each component.

The purpose of this is to restrict access of system resources to applications, processes, and users which have been given permission before hand.

The components of the trusted computing base are:

- The kernel (operating system)
- The configuration files that control system operation
- Any program that is run with the privilege or access rights to alter the kernel or the configuration files

In previous versions of AIX, and AIX 5L, a system administrator can mark trusted files as part of the Trusted Computing Base (the `chtcb` command) so that they can limit access to files other than the default root access only files.

When considering granting access of system resources to new applications, the following must be considered:

- You have fully tested the program.
- You have examined the program's code.
- The program is from a trusted source that has tested or examined the program.

The system administrator must determine how much trust can be given to a particular program. The value of the information resources on the system should be considered when deciding how much authority to grant any given application upon installation.

7.2.3.1 Trusted communication path

The trusted communication path monitors and controls user access to system commands and files, as well as to secure data passed from the user to the system that must be communicated in a secure manner, such as a password. The trusted communication path can also be used to establish a secure environment for administration and allow for secure remote system administration through the Web-based System Manager.

The trusted communication path is based on the following:

- A trusted command interpreter (`tsh` command) that only executes commands that are marked as being a member of the Trusted Computing Base
- Restricting access to a terminal to trusted programs
- A reserved key sequence, called the secure attention key (SAK), which allows the user to request a trusted communication path

7.2.4 Auditing

The object-oriented auditing subsystem provides the system administrator with a means of recording security-relevant information, which can be analyzed to detect potential and actual violations of the system security policy. The auditing subsystem has three functions, each of which can be configured by the system administrator.

7.2.4.1 Event detection

Event detection is distributed throughout the Trusted Computing Base (TCB), both in the kernel (supervisor state code) and the trusted programs (user state code). An auditable event is any security-relevant occurrence in the system. A security-relevant occurrence is any change to the security state of the system, any attempted or actual violation of the system access control or accountability security policies, or both. The programs and kernel modules that detect auditable events are responsible for reporting these events to the system audit logger, which runs as part of the kernel and can be accessed either with a subroutine (for trusted program auditing) or within a kernel procedure call (for supervisor state auditing). The information reported should include the name of the auditable event, the success or failure of the event, and any additional event-specific information that would be relevant to security auditing.

Event detection configuration consists of turning event detection on or off, either at the global (system) level or at the local (process) level. To control event detection at the global level, use the audit command to enable or disable the audit subsystem. To control event detection at the local level, you can audit selected users for groups of audit events (audit classes).

7.2.4.2 Information collection

Information collection involves logging the selected auditable events. This function is performed by the kernel audit logger, which provides both an SVC (subroutine) and an intra-kernel procedure call interface that records auditable events.

The audit logger is responsible for constructing the complete audit record consisting of the audit header, which contains information common to all events, such as the name of the event, the user responsible, and the time and return status of the event, and the audit trail, which contains event-specific information. The audit logger appends each successive record to the kernel audit trail, which can be written in either (or both) of the following two modes:

- **BIN mode** - The trail is written into alternating files providing for safety and long-term storage.
- **STREAM mode** - The trail is written to a circular buffer that is read synchronously through an audit pseudo-device. STREAM mode offers immediate response.

Information collection can be configured at both the front end (event recording) and at the back end (kernel trail processing). Event recording is selectable on a per-user basis. Each user has a defined set of audit events that are actually logged in the kernel trail when they occur. At the back end, the modes are individually configurable so that the administrator can employ the back-end processing best suited to a particular environment. In addition, BIN mode auditing can be configured to shut down the system in the event of failure.

7.2.4.3 Information processing

The operating system provides several options for processing the kernel audit trail. The BIN mode trail can be compressed, filtered, or formatted for output (or any reasonable combination of these) prior to archival storage of the audit trail. Compression is done through Huffman encoding. Filtering is done with a standard query language or SQL-like audit record selection (using the `auditselect` command) and provides for both selective viewing and selective retention of the audit trail. Formatting of audit trail records can be used to examine the audit trail, generate periodic security reports, and print a paper audit trail. The STREAM mode audit trail can be monitored in real time to provide immediate threat monitoring capability. Configuration of these options is handled by separate programs that can be invoked as daemon processes to filter either BIN or STREAM mode trails, although some of the filter programs are more naturally suited to one mode or the other.

7.2.4.4 Event selection

The set of auditable events on the system defines which occurrences can actually be audited and the granularity of the auditing provided. The auditable events must cover the security-relevant events on the system as defined previously. The level of detail you use for auditable event definition must tread a fine line between insufficient detail, which leads to excessive information collection, and too much detail, which makes it difficult for the administrator to logically understand the selected information. The definition of events takes advantage of similarities in detected events. For the purpose of this discussion, a detected event is any single instance of an auditable event. The underlying principle is that detected events with similar security properties are selected as the same auditable event.

7.2.4.5 Configuration

The auditing subsystem has a global state variable that indicates whether the auditing subsystem is on or off. In addition, each process has a local state variable that indicates whether the auditing subsystem should record information about this process. Both of these variables determine whether events are detected by the Trusted Computing Base (TCB) modules and programs. Turning TCB auditing off for a specific process allows that process to do its own auditing and not to bypass the system accountability policy. Permitting a trusted program to audit itself allows for more efficient and effective collection of information.

7.2.4.6 Information collection

Information collection addresses event selection and kernel audit trail modes. It is done by a kernel routine that provides interfaces to log information, used by the TCB components that detect auditable events and configuration interfaces, and used by the auditing subsystem to control the audit logger routine.

7.2.4.7 Audit logging

Auditable events are logged with one of two interfaces: The user state or the supervisor state. The user state part of the TCB uses the auditlog or auditwrite subroutine while the supervisor state portion of the TCB uses a set of kernel procedure calls.

For each record, the audit event logger prefixes an audit header to the event-specific information. This header identifies the user and process for which this event is being audited as well as the time of the event. The code that detects the event supplies the event type and return code (or status) and, optionally, additional event-specific information (the event tail). Event-specific information consists of object names (for example, files refused access or tty used in failed login attempts), subroutine parameters, and other modified information.

Events are defined symbolically rather than numerically. This lessens the chances of name collisions without using an event registration scheme. Also, since subroutines are auditable, the extendable kernel definition, with no fixed SVC numbers, makes it difficult to record events by number since the number mapping would have to be revised and logged every time the kernel interface was extended or redefined.

7.2.4.8 Audit record format

The audit records consist of a common header followed by audit trails peculiar to the audit event of the record. The structures for the headers are defined in the /usr/include/sys/audit.h file. The format of the information in the audit trails is peculiar to each base event and is shown in the /etc/security/audit/events file.

The information in the audit header is generally collected by the logging routine to ensure its accuracy while the information in the audit trails is supplied by the code that detects the event. The audit logger has no knowledge of the structure or semantics of the audit trails. For example, when the login command detects a failed login, it records the specific event with the terminal on which it occurred and writes the record into the audit trail using the auditlog subroutine. The audit logger kernel component records the subject-specific information (user IDs, process IDs, and time) in a header and appends this to the other information. The caller supplies only the event name and result fields in the header.

7.2.4.9 Logger configuration

The audit logger is responsible for constructing the complete audit record. You must select the audit events that you want to be logged.

7.2.4.10 Kernel audit trail modes

Kernel logging can be set to BIN or STREAM modes to define where the kernel audit trail is to be written. If BIN mode is used, the kernel audit logger must be given (prior to audit startup) at least one file descriptor to which records are to be appended.

BIN mode consists of writing the audit records into alternating files. At the start of auditing, the kernel is passed two file descriptors and an advisory maximum bin size. It suspends the calling process and starts writing audit records into the first file descriptor. When the size of the first bin reaches the maximum bin size and, if the second file descriptor is valid, it switches to the second bin and reactivates the calling process. It continues writing to the second bin until it is called again with another valid file descriptor. If, at that point, the second bin is full, it switches back to the first bin, and the calling process returns immediately. Otherwise, the calling process is suspended, and the kernel continues writing records into the second bin until it is full. Processing continues this way until auditing is turned off.

STREAM mode is much simpler than BIN mode. The kernel writes records into a circular buffer. When the kernel reaches the end of the buffer, it simply wraps to the beginning. Processes read the information through a pseudo-device called /dev/audit. When a process opens this device, a new channel is created for that process. Optionally, the events to be read on the channel can be specified as a list of audit classes.

The main purpose of this mode is to allow for timely reading of the audit trail, which is desirable for real-time threat monitoring. Another use is to create a paper trail that is written thus, preventing any possible tampering with the audit trail (if the trail is stored on some writable media).

7.2.5 Additional protection tools

There are additional tools that come standard with AIX Version 4.3 that can be used to make AIX and your applications more secure.

7.2.5.1 Physical locking

On older versions of pSeries servers a physical lock was located on the front of the system to control whether or not to boot up in normal mode, maintenance mode, or not be able to reboot at all. When IBM switched to a PCI manufacturing model for AIX hardware the decision was made to discontinue this feature, and it is only available on older microchannel based systems.

7.2.5.2 Software locking

Most people are familiar with the screen saver. It is enabled with a password and prevents anyone else from being able to use the user ID or the terminal. This comes standard with AIX in the Common Desktop Environment, which was previously explained in detail. If CDE is not installed, there can be no screen saver.

7.2.5.3 Anti-spamming feature

Starting with AIX Version 4.3.3, Sendmail has been upgraded to Version 8.9.3, which, among other things, features anti-spamming. AIX includes the necessary files to generate custom configuration files for the anti-spamming feature. While the default, /etc/sendmail.cf, does not include the anti-spamming configuration, the /usr/samples directory contains custom anti-spamming configuration files to illustrate how to configure the feature.

7.2.5.4 Console logging

AIX treats system console messages as critical system information. Previously, these messages were simply displayed on the current console device. If that screen or window was in use, the messages could be lost. Now, in addition to displaying them on the console, these messages are also logged to a file along with the originating user and the time the message was written. Now, it is possible to easily retrieve these messages, thus, improving the ability to diagnose problems and monitor system status.

In addition to this enhancement, the file system and system dump processor have also been improved so that it is easier to diagnose problems when they occur.

7.2.6 Optional protection tools

IBM and third parties have a range of software available to make an RS/6000 system even more secure than what AIX provides as standard.

7.2.6.1 IBM SecureWay Communications Server for AIX

AIX 4.3.3 offers new directory exploitation of AIX users and groups. It provides a facility which allows AIX user and group information to be optionally stored, replicated, and retrieved in an IBM SecureWay Directory for fast access (local or remote), expandability, and reliability. When an AIX system is configured, user and group related queries are sent to (and responses received from) the SecureWay Directory. All AIX user data is securely stored, replicated, and managed by the SecureWay Directory server. For a collection of AIX systems that need to share a common view of user security information, this function can significantly reduce the number of administrative operations.

IBM SecureWay Directory is an open cross-platform server optimized to support Lightweight Directory Access Protocol (LDAP) enabled applications that integrate enterprise systems. Providing a unified architecture that allows users to share data with people, applications, and network resources, the SecureWay Directory helps improve communication, speed development and deployment of Web applications, and increases the security of the network. Utilizing the power of the IBM DB2 Universal Database (UDB) and its transactional data store, the directory extends the performance and availability of DB2 to an enterprise directory service.

Note

You may only use the DB2 UDB component in association with your licensed use of the SecureWay Directory.

The SecureWay Directory (formerly, the eNetwork LDAP Directory) has been rebranded and renamed under the IBM SecureWay brand to more closely align with the IBM eBusiness portfolio for SecureWay Software. These products provide an integrated solution and a secure network platform for our customers to implement an e-business.

The new version of the SecureWay Directory steps up to the Internet Engineering Task Force (IETF) LDAP V3 support based on RFC 2251, 2252, 2253, 2254, and 2256. Many new features are provided over the eNetwork LDAP Directory, which was based on LDAP V2. LDAP V3 provides enhancements to both the LDAP protocol and the supported schema.

The new LDAP V3 protocol features include:

- **Referrals** - A list of server URL addresses are returned to a client whose request cannot be serviced. The client can use the returned server locations to continue the operation.
- **Controls** - Extension information can be added to a request for an LDAP operation.
- **Extended operation plugin support** - Additional operations can be defined for services not available elsewhere in the V3 protocol. Clients can request and receive responses with predefined syntax and semantics.

The SecureWay Directory server uses attribute-type definitions, object-class definitions, and other information called schema to determine how to match a filter or attribute value against an attribute of a directory entry. The schema matching also determines whether or not add or modify operations are permitted. The breadth of supported schema definitions has grown to support not only the schema defined by LDAP V3 but also IBM common schema and Directory-Enabled Network (DEN) schema.

Subclassing enables new object classes to be defined that inherit the object class definitions and attributes of its parent class. The new object may be defined with additional or changed attributes. Schema update operations are checked against the schema class hierarchy for consistency before being processed. Additionally, the directory permits authorized users to dynamically define new attributes and object classes to enhance the predefined directory schema.

SecureWay Directory has provided a migration utility to convert your eNetwork LDAP Directory V2.1 schema definitions to LDAP V3 format. No migration is required for the directory data. eNetwork LDAP Directory V2.1 data will work with the SecureWay Directory V3.1.1 server.

Server-specific information, such as Directory System Agent (DSA)-Specific Entry (DSE), is contained in a read-only repository, RootDSE, which contains the following information:

- Suffixes supported by the local directory server
- Distinguished Name (DN) of the subschema entries known by the server
- List of alternative (replica) servers
- LDAP version implemented by the server
- List of supported extended operations
- List of supported controls
- List of supported Simple Authentication and Security Layer (SASL) security features
- Server configuration information

SASL, which is defined in RPC 2222, is a framework for adding plugable authentication support for connection-based protocols. The directory server invokes the SASL plug-in functions to perform authentication following a bind request from a client and returns the results to the client. Two methods of authentication are supported:

- Challenge/Response Authentication Mechanism - Message Digest 5 (CRAM-MD5)
- Secure Socket Layer (SSL)

Several new features have been added to address security. In addition to certificate authentication for the server, which was available with Version 2.1, the directory now supports SSL client certificate authentication based on public keys, which provides the means to set up a protected communication channel between the client and the server. A user with a public key certificate signed by a certificate authority can use the certificate to authenticate himself or herself to the directory server:

- Full SSL Java Naming and Directory Interface (JNDI) support
- Encryption of passwords in the directory prevents passwords from being compromised via database queries or file lookup

The SecureWay Directory allows users to write server and client plug-ins, which contain additional functions that the user would like the server or client to perform. A plug-in is a dynamic link library (DLL) that can be dynamically linked with the server. The directory plug-in APIs are compatible with the Netscape Directory Server (NDS)-published APIs.

Directory clients can locate directory servers via the domain name. SecureWay Directory server addresses are published through Service Resource Records in the Domain Name Service (DNS) manually. A list of servers will be returned to the client from the DNS.

This release includes a Change Log, which logs add, delete, and modify operations to the directory server as well as changes to the change log itself. A client can access the Change Log and update its own replicated copy of the directory data by applying the changes.

Greater performance and data availability are achieved though client-side caching and server ACL Caching.

Data can be stored, retrieved, and managed in the directory using a native language code page for either single-byte or double-byte languages. Data is converted to the Universal Code Set (UCS) Transformation Format (UTF-8) character strings before being sent to and from the server. The data can be stored as either UTF-8 or as local codepage strings depending on the database configuration. This version has translated messages for Group 1 languages and Czech, Polish, Hungarian, Russian, Catalan, and Slovakian.

SecureWay Directory can be administered and configured from a Web browser-based GUI. The administrator can:

- Perform initial setup of the directory.
- Change configuration options.
- Manage the daily operations of the directory.
- User access control is provided for information stored in the directory and can be defined by an administrator. From a Web browser, users can search for or add to information in the directory. In addition, the Java-based Directory Management Tool is provided to allow a user to perform the following tasks:
  - Connect to one or many directory servers via secure or unsecure network connects
  - Browse the directory tree or directory schema

  - Add, Edit, Modify, and Delete objects, object classes, and attributes in the directory

Client access to the SecureWay Directory is supported using LDAP or HTTP protocols. AIX client applications can be developed using the enhanced elements provided for support of LDAP V3 protocols and APIs. These elements are provided by the SecureWay Client SDK, which consists of:

- Client libraries that provide a set of C-language APIs.
- C header files.
- Documentation (in the form of HTML files).
- Sample programs.
- Executable versions of the sample programs.
- Additionally, the following components are provided for developing Java applications that use Sun's JNDI. This permits Java applications to access the following LDAP-compliant directory servers:
  - JNDI class files
  - A set of class files for the LDAP service provider
  - Documentation

The LDAP libraries and utilities provided with the SDK utilize the SSL libraries, if present. The SSL libraries are provided as part of the IBM Global Security Kit (GSKit). When GSKit is installed, the LDAP library will dynamically load the SSL libraries and use them to enable support of SSL. The LDAP library is fully functional regardless of the presence of SSL. GSKit Version 3.0.1 is available on the AIX Bonus Pack 4.3.

The U.S. government's regulations regarding the export of SDKs, which provide support for strong encryption, continue to evolve. This has resulted in changes in the way IBM packages the SecureWay Directory Client SDK and the manner in which LDAP applications gain access to the strongest SSL encryption algorithms, which include 128-bit and triple DES encryption. The point of control, with respect to available levels of encryption, is now the application.

Any LDAP application that uses the SecureWay Directory Client SDK Version 3.1.1 with the required level of GSKit 3.0.1.84 (or higher) has default access to 56-bit DES encryption (over SSL). This is the case for LDAP applications (both new and existing ones) that use either the domestic or general export versions of SecureWay Client SDK Version 3.1.1.

For an LDAP application to access the stronger SSL cryptographic encryption algorithms, the application must use a new function that sets the cipher support to 128-bit/triple-DES and registers the application for the stronger cryptographic encryption algorithms. Without this function, LDAP applications have default access to a maximum of 56-bit DES encryption for SSL connections. To invoke the new function, your application must be linked with the appropriate static library that exports it. These static libraries are distributed via the IBM SecureWay Directory Security Enabler V3.1.1 package (5648-D14). For users within the U.S. and Canada, this package can be download from the following URL:

```
http://www-4.ibm.com/software/network/directory/
```

These static libraries, which provide unrestricted cipher support, and applications developed with these libraries, may be exported outside the U.S. and Canada only with the appropriate export license as provided by the U.S. government. The SecureWay Directory is Tivoli-ready.

7.2.6.2 IBM Secureway Firewall for AIX and Windows 2000

Formerly eNetwork Firewall for AIX, IBM Secureway Firewall for AIX and Windows 2000 is the next iteration of IBM’s network security product. It enables safe, secure internet access by controlling all communication leaving and going into your IP topology.

You can find more information on this product at:

```
http://www.tivoli.com/products/index/secureway_firewall/
```

7.2.6.3 Checkpoint Firewall-1

Another firewall product that runs on AIX V4.3 is Checkpoint’s Firewall-1. It is the market leader in firewall technology and has the following features:

- Patented Stateful Inspection
- OPSEC Partner Alliances
- Encryption (3DES, DES, FWZ1, 40-bit)
- Virtual Private Networks (VPNs)
- Enhanced SMP performance
- Multiple firewall synchronization
- Centralized graphical security management
- LDAP user management

More information, including a technical overview of their firewall product, can be found on the Checkpoint Firewall-1 at the following Web site:

`http://www.checkpoint.com/`

7.2.6.4 Tivoli Security Management

Tivoli Security Management is designed to allow a consistent security policy over multiple platforms. It is a centralized role-based security administration for platforms, such as UNIX, Windows 2000, AS/400, and OS/390 Security Server for RACF. It also has flexible auditing capabilities that allow you to focus on particular groups or resources.

In addition to its normal functions and productivity tools, Tivoli also offers a security engine for UNIX servers. The Tivoli Access Control Facility is an architecture that is consistent with the IBM RACF solution for OS/390. This enables you to focus your attention on security priorities, such as enterprise security policy and the protection of business resources, rather than the detailed activity of protecting specific IT assets for various platforms, applications, and databases.

Tivoli Security Management ensures that security policy is enforced consistently across both geographic and platform boundaries and also improves productivity by providing a consistent user interface and by using the Tivoli method of subscribing endpoints to a Security Profile Manager. Productivity is further enhanced by using Tivoli software to automate security tasks and allows the secure delegation of maintenance tasks to junior administrators without OS security expertise.

Tivoli Security Management is installed on the Tivoli Management Framework and exploits the Tivoli Enterprise Console for security event correlation, Tivoli Distributed Monitoring for effective security alarming, and Tivoli User Administration for efficient user account management.

For more information, visit the following Web site:

`http://www.tivoli.com/`

7.2.6.5 TCP Wrappers

Originally written by Wietse Venema, a consultant at Eindhoven University of Technology in Holland, it is estimated it is installed on as many as a million UNIX machines worldwide. TCP Wrappers keeps track of invalid access attempts and can be programmed to deny access to persons deemed suspicious (based on a history of invalid access attempts). The current version supports the System V.4 TLI network programming interface (Solaris, DG/UX) in addition to the traditional BSD sockets.

TCP Wrappers is available for free on the Internet at:

```
ftp://ftp.porcupine.org/pub/security/tcp_wrappers_7.6.tar.gz
```

7.2.6.6 Computer Oracle and Password System (COPS)

This tool is a security policy checker originally created by Dan Farmer as his final project for his undergraduate work at Purdue University. The administrator defines selected security aspects or rules to check. COPS is not a network-based checker. It must be run separately on each system.

It will report on any security weaknesses it finds, and it comes with a large documentation package that explains how to set up, run, and interpret the results. One potential drawback to COPS is that it can be run by any user of the system. Therefore, if it does exist on a system, the system administrator should be the first to run it, and he or she should do so regularly before the other users do. COPS is available for free on the Internet at:

```
http://www.fish.com/cops/
```

7.2.6.7 Security Analysis Tool for Auditing Networks (SATAN)

This highly popular tool, originally created by Wietse Venema and Dan Farmer, is a network-based security checker. It is more or less a following product to COPS with a network-based implementation. It has received much attention recently for its ability to quickly spot security flaws across a network.

Since SATAN is network-based, it basically attacks systems by trying to exploit security holes and weaknesses. Of course, SATAN's aim is to report what it finds versus actually exploiting a vulnerability. One system running SATAN can monitor security for thousands of systems on a network.

SATAN is available for free on the Internet at:

```
ftp://ftp.porcupine.org/pub/security/satan-1.1.1.tar.Z
```

7.2.6.8 Stalker

Haystack Labs has a tool, known as Stalker, that can be purchased from them. It is designed to detect and respond to system misuse by comparing logs of system events against a database of known ways to break into a UNIX system. If it finds that tampering has occurred, it sends an alarm via e-mail, page, SNMP, or to a printed report identifying who did what, when, where, and how.

At the Information Security Award Event in 1996, Stalker won *Best Unix Security Product*. More information can be found at the following Web site:

`http://www.haystack.com/`

7.3 Windows 2000 security

There have been many improvements to Windows security with the release of Windows 2000. The primary features of the Windows 2000 security model are user authentication and access control, but there are many new parts that make up the workings for these. Perhaps most noteworthy is the introduction of the directory service, Active Directory. This is the new foundation of Windows 2000 and affects the way many components work.

7.3.1 Security Configuration Manager

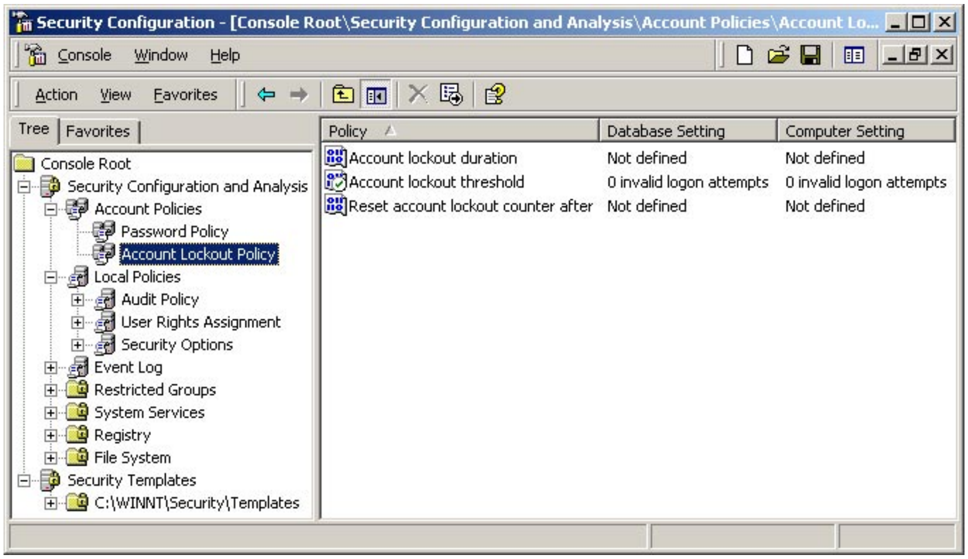
The Security Configuration Manager is designed to be the one-stop security configuration and analysis tool. It allows network administrators to set up a template that can set security-sensitive registry settings, access controls on files and registry keys, and define security settings for system services. Once this has been done, the template can be applied to many computers in one operation.

This tool also allows the setup of security policies. Other operations include access control, group membership, event log, Internet Protocol Security (IPSec), and Public Key policies.

To start the Security Configuration and Analysis tool, it must be added to an MMC console.

Start the MMC by choosing the Run command from the Start menu, type in `MMC` and click **OK**.

From the console menu, choose **File** -> **Add/Remove Snap-In**, click **Add**, and then select the **Security Configuration and Analysis** and **Security Templates** snap-ins. Save the console and it will appear in the Administrative tools folder. Figure 55 on page 188 shows the window for Security Configuration and Analysis:



Policy	Database Setting	Computer Setting
Account lockout duration	Not defined	Not defined
Account lockout threshold	0 invalid logon attempts	0 invalid logon attempts
Reset account lockout counter after	Not defined	Not defined

Figure 55. Console for security configuration and analysis

7.3.2 Active Directory

Active Directory is one of the major added functionalities in Windows 2000. It is a directory service acting as a repository for all domain and account information as well as domain security policies. It is an improvement over the previous technology with new additions and enhancements that allows an object on a network to be tracked and located.

The Active Directory allows administrators to manage user accounts and access rights from a central location, grant or deny access rights, and delegate security administration. According to the Microsoft white paper, *Active Directory Technical Summary*, the Active Directory “is secure, distributed, partitioned and replicated. It is designed to work well in any size installation, from a single server with a few hundred objects to thousands of servers and millions of objects.”

The data model for the Active Directory is derived from the X.500 model. There are some protocols in the X.500 model that the Active Directory has intentionally chosen not to follow; it does follow the LDAP and MAPI-RPC protocols. The Active Directory is part of the Windows 2000 Trusted Computing Base and is a full participant in Windows 2000 security.

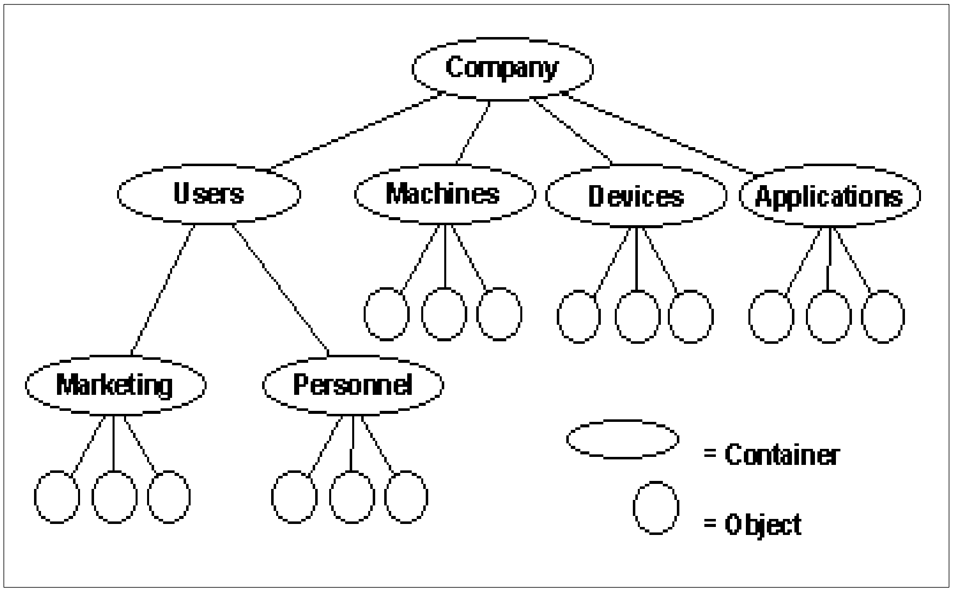


Figure 56. Active Directory object storage

Objects are stored in a hierarchical, object-oriented manner, and the Active Directory provides multi-master replication to support distributed network environments. Figure 56, from Microsoft's *Active Directory Overview*, shows a graphical representation of how the Active Directory stores objects.

Containers are used to represent a collection of related objects.

Given that a domain is a single security boundary of a computer network, the Active Directory could be considered to be made up of one or more domains. On a standalone workstation, the domain is the computer itself, whereas in a network, a domain can span several physical locations.

One or more directory partitions make up the Active Directory. Directory partitions are contiguous subtrees of the directory that form a unit of replication. This means that any given replica is always a replica of some directory partition.

The Global Catalog (GC) holds a replica of every object in the Active Directory but only a scaled down version with a small number of each object's attributes (those most often searched for). This allows users and applications to find objects in an Active Directory domain tree without knowing what domain it belongs to. This Global Catalog is built automatically by the Active Directory replication system.

One of the most important security features of the Active Directory is delegation. Rather than having domain administrators with complete authority over large segments of the user population, delegation allows a higher administrative authority to grant specific administration rights to highly-trusted individuals or groups. Windows 2000 defines many specific permissions and user rights for this purpose. Using a combination of group membership and permissions, the most appropriate role for a person can be defined.

Figure 57 shows an example of the management of network resources with the Active Directory.

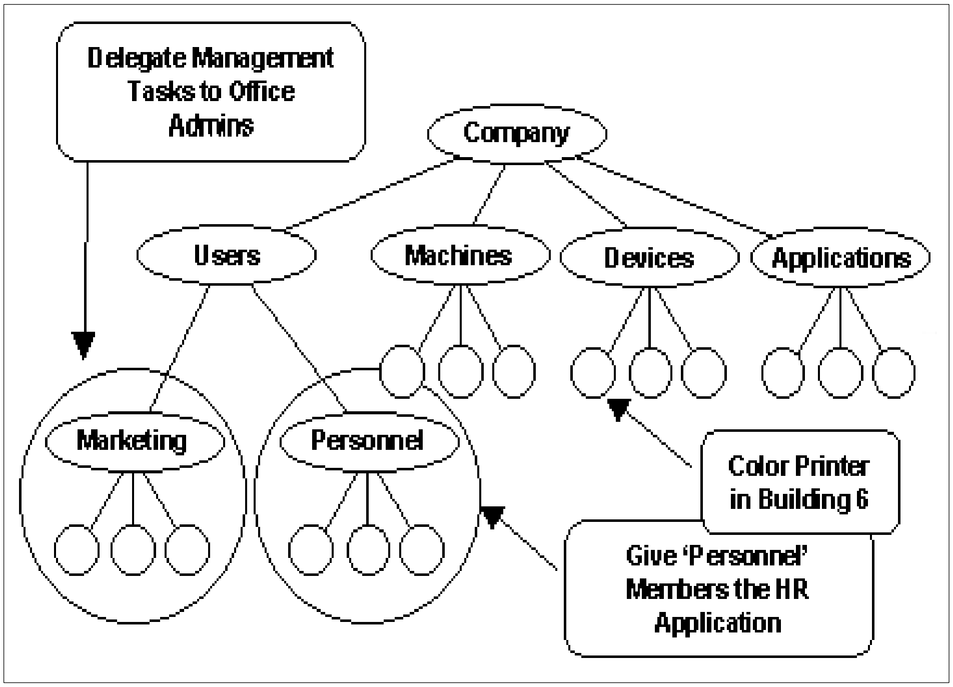

Delegate Management Tasks to Office Admins
Company
Users
Machines
Devices
Applications
Marketing
Personnel
Color Printer in Building 6
Give 'Personnel' Members the HR Application

Figure 57. Managing network resources with the Active Directory

Examples of specific permissions that might be delegated by the administrator are things like resetting users passwords or creating a new user.

With a number of different authentication methods, Active Directory provides an increased level of security for Windows 2000 systems. Once a user is logged on, all of the system resources are protected through a single authorization model. Figure 58 on page 191 shows an example of how a system might be set up with different authentication methods.

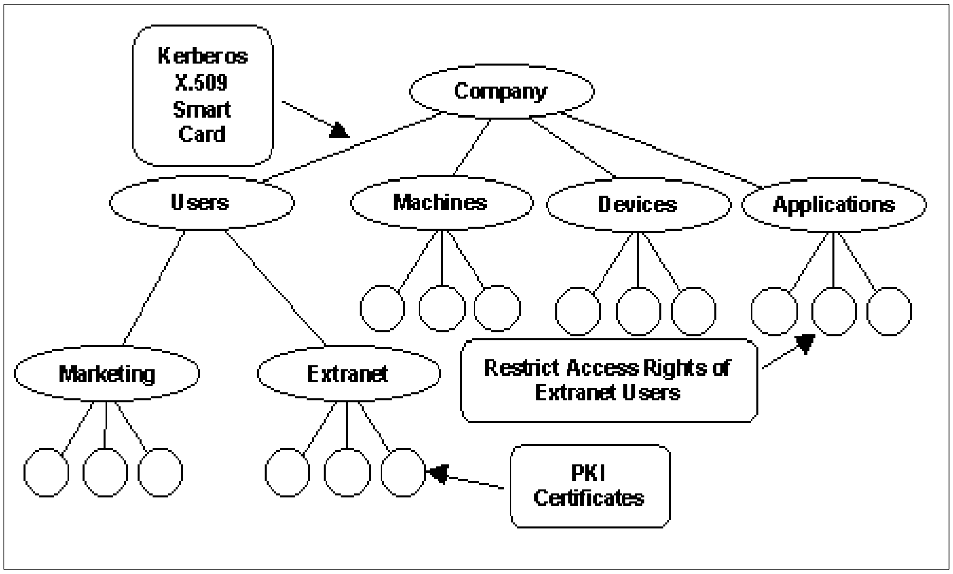


Figure 58. Active Directory security example

7.3.3 Logon process

In order to log into a system or a domain for the Windows 2000 Server packages, the user must press the Ctrl+Alt+Del key sequence to display the Logon Information dialog box. This key sequence prevents against any application running in the background, such as a Trojan Horse, that attempts to capture the user's logon information.

The user must then enter the username and password and specify whether to log on locally or to a particular domain.

The system will check in the Active Directory to see if the user exists and if the password provided is valid. It will reject the connection if one of the entries is invalid without saying if the user exists or if the password is correct (this is done intentionally).

Windows 2000 determines if the user has an account and if the password is valid and then checks what groups the user is a member of. The security subsystem creates an access token that represents the user. It contains information, such as the user's Security ID (SID), the username, and the groups to which the user belongs. This token is sent to any computer that the user accesses.

This access token (or a copy of it) is associated with each process started by the user. The access token and process association are called a subject. When accessing a resource (file, directory, and so on), the content of the subject is checked against the Access Control List of the object being accessed by an access validation routine. This determines the users rights and permissions on that computer.

7.3.3.1 User profile and home directory

Like AIX, Windows 2000 can set up a profile for each user and a home directory in which to store personal files.

A user profile is a file or directory with a collection of files containing information about the user's environment. The user profile is loaded each time the user logs on. Modifications to the environment made by the user are saved in that profile when the user logs off.

There are actually different types of profiles: Mandatory user profiles, personal user profiles, user default profiles, and system default profiles. A mandatory user profile can be set by the system administrator to users in a domain. This profile defines which environment settings must be set when a user logs on as well as which application must be started and which network drive must be connected. If a user makes any changes to the environment, these changes will not be saved when the user logs off.

A personal user profile (when assigned to a user) can be modified by the user, and changes are saved when the user logs off. The user will retrieve the previous environment after login.

The user default profile is the standard Windows 2000 default profile that is used when a user account has not been assigned a profile or when a user has never logged onto the system. Also, if a user profile cannot be accessed when a user logs in, the user default profile is assigned.

The system default profile is the one that appears when nobody is logged in (when the Ctrl+Alt+Del dialog box is displayed).

It is also possible for an administrator to set up a logon script that executes after a user logs in. It may be used to establish a network connection, configure the environment, or start up a specific application.

7.3.4 User authentication and authorization

There are two types of user authentication with the Windows 2000 security model. The interactive logon confirms the user's identification to the Active Directory or the user's local computer. Network authentication confirms the user's identification to any network service that the user is attempting to access. These authentications can be done using Kerberos V5 authentication, Secure Sockets Layer/Transport Layer Security (SSL/TLS), and, to remain backward compatible with Windows NT Version 4.0, NTLM (Windows NT LAN Manager) authentication. Kerberos is the replacement for NTLM as the primary security protocol in Windows 2000.

The model for implementing access control is through authorization. Once the authentication has been received for a user account, that user can access an object according to the access granted via the user's rights or the permission of the object. The owner of that object (by default, the creator) should set the permissions for an object.

Each user account in Windows 2000 has a number of security-related options that determine how someone logging on with that ID is authenticated on the network. There are a number of password options that can be set, namely:

- User must change password at next logon
- User cannot change password
- Password never expires
- Store passwords using reversible encryption (used for users logging on from Apple computers)
- Enforce password history (how many passwords are remembered)
- Minimum/maximum password age
- Complexity requirements

Other security options include:

- Smartcard required for interactive logon
- Account trusted for delegation
- Account is sensitive and cannot be delegated
- Use DES encryption types for this account
- Do not require Kerberos pre-authentication

In the Active Directory Users and Computers application, there are built-in groups that Windows 2000 provides as well as predefined users and groups. The predefined users are the Administrator and Guest users. These are designed primarily for the initial logon and configuration of a local computer. ACLs, Access Control Entries (ACEs) and file and folder permissions are also used to maintain security levels.

The Active Directory Users and Computers application is designed to allow user accounts, computer accounts, security and distribution groups, and published resources to be added, deleted, and modified within an organization's directory. For Windows 2000 Professional, this comes as an optional administration package that enables you to administer Active Directory from a computer that is not a domain controller.

7.3.4.1 Built-in groups

The following groups are built-in to Windows 2000 Server:

- Account Operators
- Administrators
- Backup Operators
- Guests
- Print Operators
- Replicator
- Server Operators
- Users

These built-in groups are all of the Security Group Built-in Local type. This means they have domain local scope and are used to assign default sets of permissions to users who will have some administrative control over that domain.

If the machine is a stand-alone machine, only local user accounts can belong to a local group. If the machine is a member of a domain, the local group can contain local user accounts, domain user accounts, trusted domain user accounts, and global groups from the domain or from a trusted domain.

Global groups are defined at the domain level and can be exported to remote domains. Members of global groups are domain user accounts or trusted domain user accounts. A local group cannot be a member of a global group.

7.3.4.2 Predefined groups

The following groups are predefined in Windows 2000 Server and placed in the Users folder for Active Directory Users and Computers:

- Cert Publishers
- Domain Admins
- Domain Computers
- Domain Controllers

- Domain Guests
- Domain users
- Enterprise Admins
- Group Policy Admins
- Schema Admins

These predefined groups are all of the Security Group Global type and can be used to collect the various types of user accounts in that domain into groups. From there, it is possible to put these groups in groups with domain local scope in both that domain and in others.

7.3.4.3 Special identities

The Windows 2000 Server packages include three special identities in addition to the built-in and predefined groups. They do not have any specific memberships that can be modified or viewed, but they can represent different users at different times. Users are automatically assigned to these special identities whenever they log on or access a particular resource. The following identities are, generally, referred to as groups:

- **Everyone** - This means all current network users, including guests and users from other domains. Users are automatically added to this group when they log on to the network.
- **Network users** - This includes all users accessing a given resource over the network (as opposed to a locally accessed resource). Users are automatically added to this group when they access a given resource over a network.
- **Interactive users** - This includes all users currently logged on to a particular computer and accessing a given resource located on that computer (opposite of network). Users are automatically added to this group when accessing a given resource on the computer on which they are logged in.

7.3.4.4 User rights and privileges

Administrators can assign specific rights to either group accounts or individual accounts. User rights apply to user accounts, and privileges apply to objects. Table 13 on page 196 shows the default rights for the built-in groups discussed previously and the groups assigned those rights by default:

Table 13. Default rights of built-in user groups

User right	Groups assigned by default
Access this computer from the network	Administrators, Everyone, Power Users
Back up files and folders	Administrators, Backup Operators
Bypass traverse checking	Everyone
Change the system time	Administrators, Power Users
Create a pagefile	Administrators
Debug programs	Administrators
Force shutdown from a remote system	Administrators
Increase scheduling priority	Administrators, Power Users
Load and unload device drivers	Administrators
Log on locally	Administrators, Backup Operators, Everyone, Users, Guests and Power Users
Manage auditing and security log	Administrators
Modify firmware environment variables	Administrators
Profile single process	Administrators, Power Users
Profile system performance	Administrators
Restore files and folders	Administrators, Backup Operators
Shut down the system	Administrators, Backup Operators, Everyone, Power Users and Users
Take ownership of files and other objects	Administrators

7.3.4.5 Access control entries and lists

Similarly to AIX, Windows 2000 has Access Control Lists (ACLs). An ACL is a list of Access Control Entries (ACEs) that allow or deny access rights to individuals or groups. It is stored as a binary value called a security descriptor.

ACLs protect all objects in the Active Directory by determining who can see an object and what actions each user can perform on it. A user who does not have access to a particular object will not even know that it is there.

There is a Security Identifier (SID) in each ACE. Its purpose is to identify the principal (user or group) to which the ACE applies and information about what access the ACE will either allow or deny.

Directory object ACLs contain two lots of ACEs. There are ACEs that apply to the whole object and others that apply to the individual attributes of that object. The purpose of this is to allow the administrator to control which users can see an object and what properties of that object they can see.

7.3.4.6 File and folder permissions

Access to network files and folders is controlled with permissions. With the Windows 2000 security system, you specify which users can use which files, folders, and shares and how they can be used.

Each permission for either a file or a folder consists of a logical group of special permissions. Those special permissions are as follows:

- **Traverse Folder/Execute File** - Grants or denies permission to move through folders and to reach other files or folders, even if the user has no permissions for the traversed folders (this applies to folders only). Traverse Folder takes effect only when the group or user is not granted the Bypass traverse checking user right in the Group Policy snap-in. (By default, the Everyone group is given the Bypass traverse checking user right).
- **Execute File** - Grants or denies permission to run program files (this applies to files only).
- **List Folder/Read Data** - Grants or denies permission to view file names and subfolder names within the folder (this applies to folders only).
- **Read Data** - Grants or denies permission to view data in files (this applies to files only).
- **Read Attributes** - Grants or denies permission to view the attributes of a file or folder, such as read-only and hidden. Attributes are defined by NTFS.
- **Read Extended Attributes** - Grants or denies permission to view the extended attributes of a file or folder. Extended attributes are defined by programs and may vary by program.
- **Create Files/Write Data** - Grants or denies permission to create files within the folder (applies to folders only).
- **Write Data** - Grants or denies permission to make changes to the file and overwriting existing content (applies to files only).
- **Create Folders/Append Data** - Grants or denies permission to create folders within the folder (this applies to folders only).

- **Append Data** - Grants or denies permission to make changes to the end of the file but not changing, deleting, or overwriting existing data (this applies to files only).
- **Write Attributes** - Grants or denies permission to change the attributes of a file or folder, such as read-only or hidden. Attributes are defined by NTFS.
- **Write Extended Attributes** - Grants or denies permission to change the extended attributes of a file or folder. Extended attributes are defined by programs and may vary by program.
- **Delete Subfolders and Files** - Grants or denies permission to delete subfolders and files, even if the Delete permission has not been granted on the subfolder or file.
- **Delete** - Grants or denies permission to delete the file or folder. If you do not have Delete permission on a file or folder, you can still delete it if you have been granted Delete Subfolders and Files on the parent folder.
- **Read Permissions** - Grants or denies reading permissions of the file or folder, such as Full Control, Read, and Write.
- **Change Permissions** - Grants or denies changing permissions of the file or folder, such as Full Control, Read, and Write.
- **Take Ownership** - Grants or denies permission to take ownership of the file or folder. The owner of a file or folder can always change permissions on it, regardless of any existing permissions that protect the file or folder.
- **Synchronize** - Grants or denies permission for different threads to wait on the handle for the file or folder and synchronize with another thread that may signal it. This permission applies only to multi-threaded, multi-process programs.

The permissions for files are:

- Full Control
- Modify
- Read & Execute
- Read
- Write

Table 14 shows which file permission is associated with each special permission:

Table 14. File permissions

Special Permissions	Full Control	Modify	Read & Execute	Read	Write
Traverse Folder / Execute File	X	X	X		
List Folder / Read Data	X	X	X	X	
Read Attributes	X	X	X	X	
Read Extended Attributes	X	X	X	X	
Create Files / Write Data	X	X			X
Create Folders / Append Data	X	X			X
Write Attributes	X	X			X
Write Extended Attributes	X	X			X
Delete Subfolders and Files	X				
Delete	X	X			
Read Permissions	X	X	X	X	X
Change Permissions	X				
Take Ownership	X				
Synchronize	X	X	X	X	X

Folders have the same permissions as files, with one additional permission to allow a user to list folder contents. Table 15 shows which folder permission is associated with each special permission:

Table 15. Folder permissions

Special Permissions	Full Control	Modify	Read & Execute	List Folder Content	Read	Write
Traverse Folder / Execute File	X	X	X	X		
Traverse Folder / Execute File	X	X	X	X	X	
List Folder / Read Data	X	X	X	X	X	
Read Attributes	X	X	X	X	X	
Read Extended Attributes	X	X				X
Create Files / Write Data	X	X				X
Create Folders / Append Data	X	X				X
Write Attributes	X	X				X
Write Extended Attributes	X					
Delete Subfolders and Files	X	X				
Delete	X	X	X		X	X
Read Permissions	X			X		
Change Permissions	X					

Special Permissions	Full Control	Modify	Read & Execute	List Folder Content	Read	Write
Take Ownership	X	X	X		X	X
Synchronize	X	X	X	X	X	X

7.3.5 Auditing

Windows 2000 allows the monitoring of security-related events. A security log is generated so that the events can be reported. It is also possible to generate an audit trail to track the security administration events on the system.

Before auditing is implemented, an auditing policy must be decided upon. This is to identify what you want to audit. You can choose to audit the success or failure of the following categories:

- Audit account logon events
- Audit account management
- Audit directory service access
- Audit logon events
- Audit object access
- Audit policy change
- Audit privilege use
- Audit process tracking
- Audit system events

By default, these are all turned off when Windows 2000 is first installed.

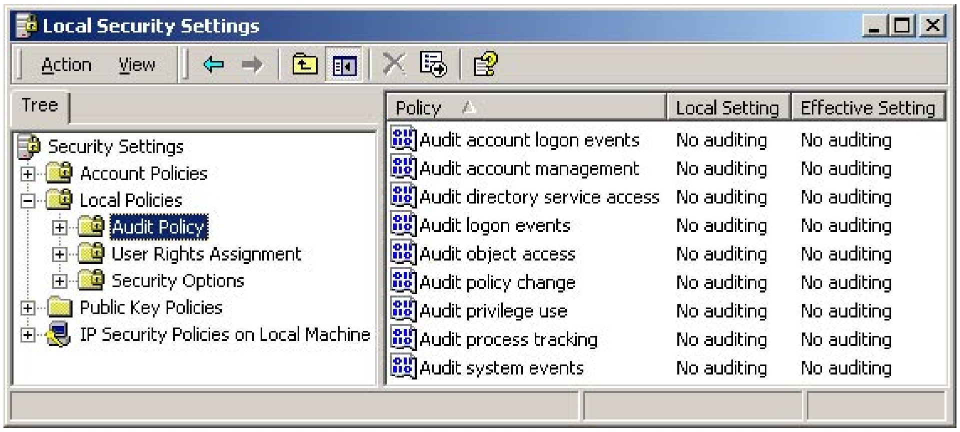
Auditing a local object will create an entry in the security log. The entries that appear in this log will depend on the auditing categories selected for your auditing policy. Although setting up the auditing policy has changed since Windows NT Version 4.0, the security log is still viewed with the Event Viewer. The following events can be audited:

- System restart
- System shutdown
- Authentication package loading
- Registered logon process
- Audit log cleared
- Number of audits discarded
- Logon successful
- Unknown user name or password

- Time restricted logon failure
- Account disabled
- Account expired
- Invalid workstation
- Logon type restricted
- Password expired
- Failed logon
- Logoff
- Open object
- Close handle
- Assign special privilege
- Privileged service
- Privileged object access
- Process created
- Process exit
- Duplicate handle
- Indirect reference
- Privilege assigned
- Audit policy change
- Domain changed
- User changed
- User created
- User deleted
- Global group member removed
- Global group member added
- Domain local group changed
- Domain local group created
- Domain local group member removed
- Domain local group member added
- Domain local group member deleted

As an example, we will enable auditing of failed logon attempts on a Windows 2000 Professional machine.

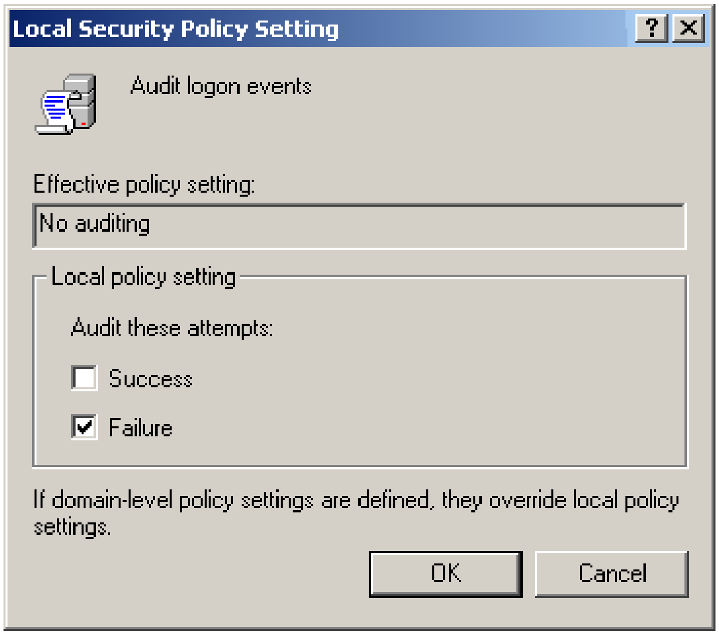
Start the Local Security Policy console, located in the Administrative Tools folder, and navigate your way down to the Audit policies as seen in Figure 59 on page 203.



Policy	Local Setting	Effective Setting
Audit account logon events	No auditing	No auditing
Audit account management	No auditing	No auditing
Audit directory service access	No auditing	No auditing
Audit logon events	No auditing	No auditing
Audit object access	No auditing	No auditing
Audit policy change	No auditing	No auditing
Audit privilege use	No auditing	No auditing
Audit process tracking	No auditing	No auditing
Audit system events	No auditing	No auditing

Figure 59. Audit Policy

Double click on the **Audit logon events** entry and check the failure box to enable auditing of only failed logon attempts, as seen in Figure 60.



Audit logon events

Effective policy setting:

No auditing

Local policy setting

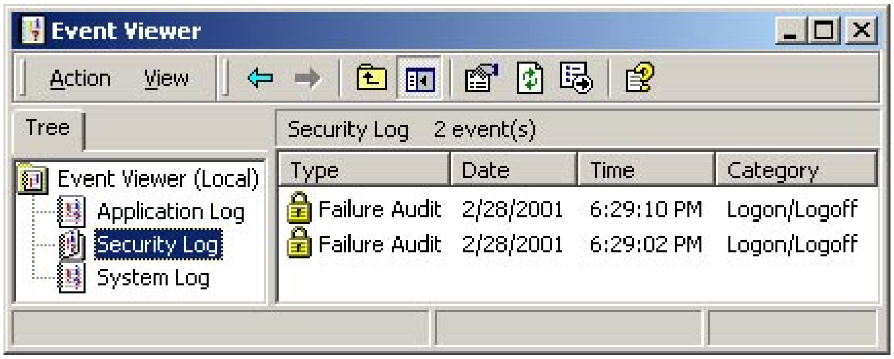
Audit these attempts:

☐ Success

☑ Failure

If domain-level policy settings are defined, they override local policy settings.

Figure 60. Logon policy setting



Security Log 2 event(s)

Type	Date	Time	Category
Failure Audit	2/28/2001	6:29:10 PM	Logon/Logoff
Failure Audit	2/28/2001	6:29:02 PM	Logon/Logoff

Figure 61. Event Viewer displaying failed logon attempts

If we log off and try to log on again using an incorrect password, we can examine the Security Log in the Event Viewer for entries pertaining to our logon attempts. The two failed attempts can be seen in Figure 61.

7.3.6 Additional protection tools

Windows 2000 has a security menu that a logged on user can access by pressing Ctrl+Alt+Del. From this menu, a user can select the following buttons:

- **Lock Workstation** - This function is a part of Windows 2000, not an applet running on top of the operating system. Once the workstation is locked, the username and password of the user who began the session must be entered to unlock it. The Administrator can also unlock the workstation. When the workstation is locked, the system screen saver is invoked after a specified amount of time.
- **Logoff** - This option closes all open applications, logs off the user, and displays the logon dialog box. Server services are not stopped and continue to run even if the user has logged off (for example, network resources are still being shared).
- **Shut Down** - This option prepares the machine to be turned off or reset.
- **Change Password** - This option allows the user to change the current password.
- **Task Manager** - This option starts the Windows Task Manager, which allows the user to display all currently-running tasks and processes and perform actions on these tasks or processes.

As with most systems, Windows 2000 has a screen saver function that can be activated after a specified period of idle time. It can optionally be password protected to provide an additional level of security.

7.3.7 Optional protection tools

There are tools that are made available from third parties that allow the security levels on the Windows 2000 systems to be increased above what is provided as standard.

7.3.7.1 Smart cards

Smart cards are already supported as part of the public key infrastructure (PKI) that has been integrated into Windows 2000. This provides a portable and tamper-resistant storage for protecting private keys, account numbers, passwords, and other forms of personal information. They also enhance software-only solutions, such as client authentication, logging on to a Windows 2000 domain, system administration, secure storage, code signing, and securing e-mail.

A public key infrastructure (PKI) is a method of using digital certificates, certification authorities (CAs), and other registration authorities that verify/authenticate each user involved in an electronic transaction through the use of public key cryptography. The PKI in Windows 2000 is based on X.509 and lets organizations issue public-key certificates for user authentication without depending on CA services.

By using smart cards, a user must log in using the card and personal identification number (PIN). If the PIN is entered incorrectly a number times in a row, the smart card is locked. For someone to try to hack into that person’s account, they would need to obtain the physical card and the PIN. This makes it more secure than a user name and password, which is more prone to attacks.

7.3.7.2 IBM eNetwork Firewall for AIX and Windows 2000

As mentioned previously in 7.2.6.1, “IBM SecureWay Communications Server for AIX” on page 179, the eNetwork firewall software is also available on Windows NT. While this product has not been ported to Windows 2000 at this time, it is expected to be done in the near future.

7.3.7.3 WebTrends Corporation

WebTrends offers two security products for the Windows 2000 platform. The first is WebTrends Firewall suite. This product manages, monitors, and reports on firewall activity in real time and is compatible with most firewall software available on the market. There is also the WebTrends Security

Analyzer, which is designed to help discover and fix the latest known security vulnerabilities on your Internet, intranet, or extranet systems. Analysis can be on-demand or scheduled, and custom scripts can be created for vulnerability tests.

For more information on these products, visit the WebTrends Web site at:

`http://www.webtrends.com/`

Chapter 8. System management

This chapter covers the base operating system management functions included in AIX 5L and Windows 2000. This chapter focuses on utilities and tools that are provided with the system. It will not mention systems management products, such as Tivoli for AIX or Microsoft System Management Server (SMS) for Windows 2000.

8.1 AIX 5L System Management

In this section, we describe information for understanding the tasks that you perform as an AIX system administrator, as well as the tools provided for system management.

8.1.1 AIX 5L installation methods

Installation involves not only installation of the operating system on one machine but also network installation, product installation, product maintenance, and operating system maintenance.

This section explores the basic methods available for installing the AIX 5L operating system as well as for upgrading and managing levels of the operating system. These methods range from a completely new installation to upgrading an existing installation on one or more systems, automatically, throughout a network. This is not a “how-to” chapter by any means. Instead, only the basic installation and management concepts are introduced. For more information on how to install AIX 5L, you can refer to the following books:

- *AIX Version 5L Version 5.1 Installation Guide*, SC23-4112
- *AIX Version 5L Version 5.1 Network Installation Management Guide and Reference*, SC23-4113

These manuals are shipped with AIX 5L in hardcopy format and in softcopy format as a part of Base Documentation CD.

AIX provides a rich set of installation and management options. This comes partly from its heritage; AIX was, traditionally, a server or a highly-technical workstation. In this market, system administrators tend to tailor a system for a specialized environment, installing only software that is actually needed to perform the required task for each system. By default, the AIX installation process installs only the minimum portion of the operating system needed for system operation. The system administrator can then install additional parts of the operating system depending on the system in use.

On the other hand, an administrator may want to remove selected software or parts of a software package. AIX allows for clean deinstallation of software as well. This is not to say that AIX is difficult to install or maintain in any way. What it means is that you have choices. These choices, along with many other facilities, are a direct response to customer requirements and feedback.

8.1.1.1 AIX 5L installation choices

AIX 5L can be installed from CD-ROM or from the network if you already have one AIX system configured as the Network Installation Manager (NIM) Server, from the network. The NIM will be described in section 8.1.1.3, “Product installation interfaces” on page 212.

There are three methods for installing the Base Operating System (BOS) as listed in the following:

New and complete overwrite

This is a typical Base Operating System (BOS) installation in which all the default options are selected. This is the option users will select when installing a completely new system or when they want to overwrite an existing operating system and start from scratch.

If you are overwriting an existing AIX installation, and if AIX is installed on disks containing the existing root volume group (rootvg), user-defined volume groups are preserved and are still there after the installation. These volume groups need to be imported and the file systems mounted in order to access the data on them.

The system will reboot at the end of the BOS installation, and the user will be presented with the Installation Assistant to perform basic customizing and to continue installing additional software if desired.

Preservation installation

This installation method is used when a version of BOS is installed on your system and you only want to preserve the user data in the root volume group. This method overwrites the /usr, /tmp, /var, and / (root) file systems by default; so, any user data in these directories is lost. However, there is a file named /etc/preserve list, which is a simple ASCII file that can be used to preserve additional files if so desired.

User-defined volume groups (other than rootvg) will be preserved and activated automatically after the Preservation Install. System configuration needs to be done after a Preservation Install because, if not explicitly preserved, all files in the /etc directory, which resides in / file system will be overwritten. The Installation Assistant screen will appear at the completion of the Preservation Install.

Migration installation

This installation method is used to upgrade an earlier version of the BOS to AIX 5.1 (for example, to upgrade from AIX 4.3). This method preserves all file systems except /tmp, including the root volume group, logical volumes, and system configuration files.

Installation made from system backup media (mksysb)

The `mksysb` command allows the system administrator to entirely back up the root volume group and create a bootable tape or CD-ROM. The system can then be restored by booting from this media and restoring `rootvg`. This is known as a mksysb installation or *cloning* the system.

Note

> If we create a bootable system image (mksysb) on one system and want to install (restore) it on the other system, great attention must be paid to the the systems (single processor/symmetrical multi processors), the adapters installed, and the external devices installed (on both systems)!

Network installation

A system administrator can use any of the above methods to install a system from the network by using a facility called Network Installation Management (NIM), which will be described in “Network Installation manager (NIM)” on page 217.

8.1.1.2 AIX 5L installation process

A key design goal for AIX product packaging was to break AIX into small installable units. In AIX, the smallest installable unit is called a file set and is, basically, a collection of files. Note that, in AIX, we cannot install or uninstall one particular file. This allows the installation process to install a minimum operating system environment and then allows the system administrator to install additional required parts of the operating system’s software.

The installation process automatically detects what type of system we are installing, either Uni-Processor or Symmetrical Multi Processor, and installs the appropriate operating system Kernel that is the core of any Unix operating system. Note that a uniprocessor system can work with a multi processor kernel, although it is slightly slower, but a Symmetrical Multi Processor Systems cannot work with a uni-processor kernel.

The installation process also detects all installed adapters in your system and installs all the required device drivers for them.

Note

If we have external devices, such as external tape drives, and they are not powered on during installation, the installation process cannot detect them and will not automatically install the required device drivers (in this example, the tape device driver).

In this case, the system administrator must later manually install the required file sets with the particular device drivers.

In the first stage of installation, the installation process sends a message to the *system console* attached to serial port S1 (if connected) and to all terminals/monitors attached to graphical cards (if installed). The system administrator has the ability to choose one of these devices to be configured as a system console.

If the installation process detects a graphic adapter installed in the system, it will also install X Window and CDE (Common Desktop Environment) software.

AIX install process

The BOS boot process is used when booting the system from a bootable media, usually CD-ROM or tape. The menus are optimized to provide a very quick installation process when default values are accepted. However, the user can modify many of the options with only a few simple selections from prompted menus. A typical AIX installation only takes 20 minutes (this time variable, of course, depends on the performance of your system). If we are using NIM for installation, there is even a no-keystroke option available. Figure 62 on page 211 describes the installation process.

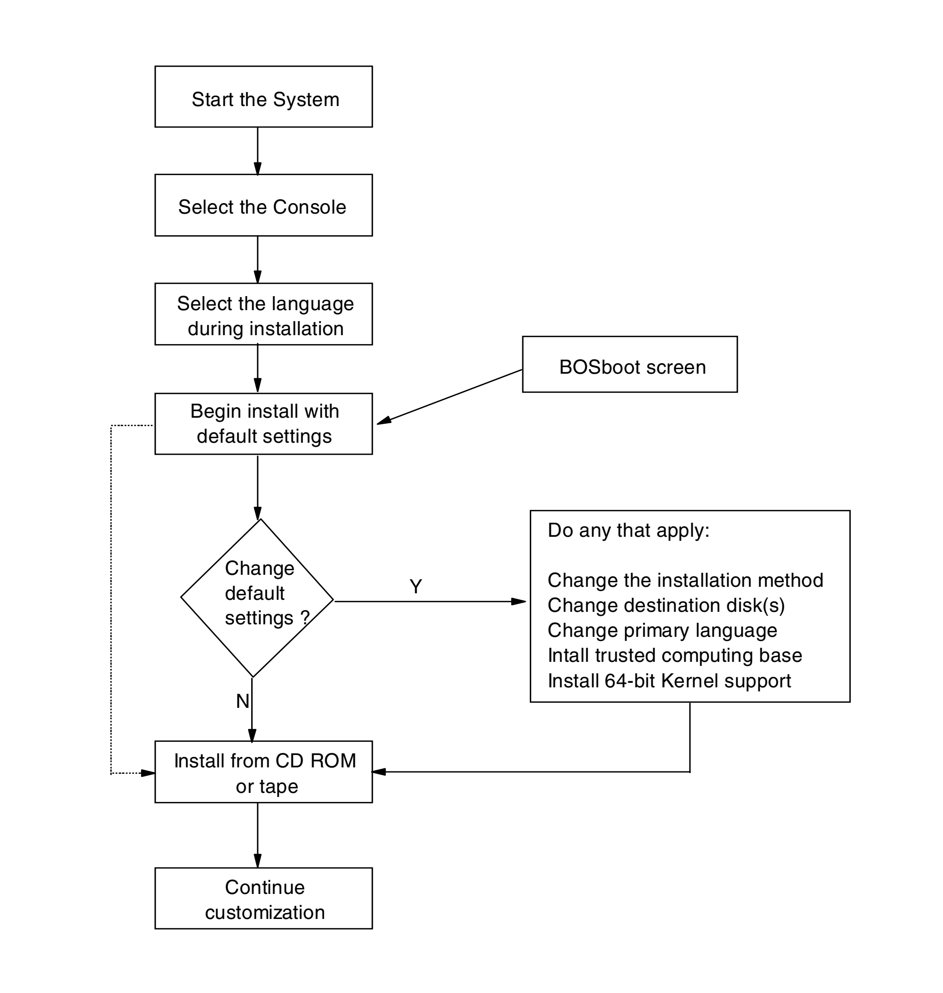
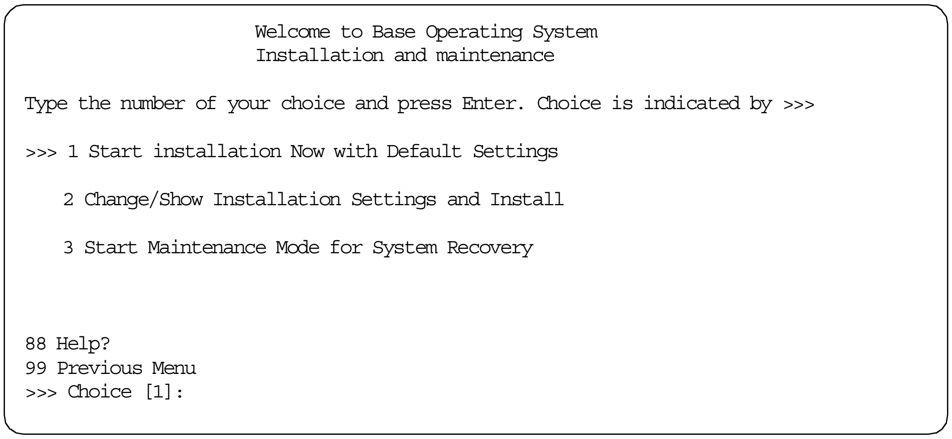

Start the System
Select the Console
Select the language during installation
BOSboot screen
Begin install with default settings
Change default settings ?
Y
Do any that apply:
Change the installation method
Change destination disk(s)
Change primary language
Intall trusted computing base
Install 64-bit Kernel support
N
Install from CD ROM or tape
Continue customization

Figure 62. AIX installation process

The AIX BOS boot screen, shown in Figure 63 on page 212, also contains an option that allows the administrator to get into a maintenance shell and perform isolated repairs in AIX. The use of this maintenance shell is somewhat restricted. However, if it is possible to activate the root volume group, full AIX functionality can be achieved. The maintenance shell is, typically, used as an emergency repair path for resolving some serious system problems. Normal or routine system maintenance does not require the use of this maintenance shell.



```
                  Welcome to Base Operating System
                  Installation and maintenance

Type the number of your choice and press Enter. Choice is indicated by >>>

>>> 1 Start installation Now with Default Settings

    2 Change/Show Installation Settings and Install

    3 Start Maintenance Mode for System Recovery



88 Help?
99 Previous Menu
>>> Choice [1]:
```

Figure 63. AIX BOS boot screen

8.1.1.3 Product installation interfaces

Once the basic (minimum) installation is done, there are five user interfaces available to install or update AIX: Command line user interface, System Management Interface Tool (SMIT), Configuration assistant, Web-based System Manager and Network Installation Management (NIM).

The most useful ones are the command line interface and SMIT.

From all four user interfaces, we can also use the Network Installation Manager (NIM) to install our system through the network (from the NIM server).

Command line user interface

The AIX command line is probably the most powerful interface on an AIX system. It is the original system interface on any UNIX system. In fact, most GUI-based interfaces are more or less just a front-end to the command line. The ability to perform actions with point-and-click GUIs and also from the command line is what makes AIX system management flexible. By having a good command line interface, an administrator can program commands via the shell to automate or repeat desired actions.

The most important/used command for software installation is the `installp` command.

SMIT

The System Management Interface Tool (SMIT) is the system management interface workhorse of AIX. One of the many sub-functions of SMIT is software maintenance. The SMIT panels are very simple to follow and use. SMIT has both a graphical version called with the `smit` command and an ASCII (streams) version called with smitty. Both commands provide identical functionality.

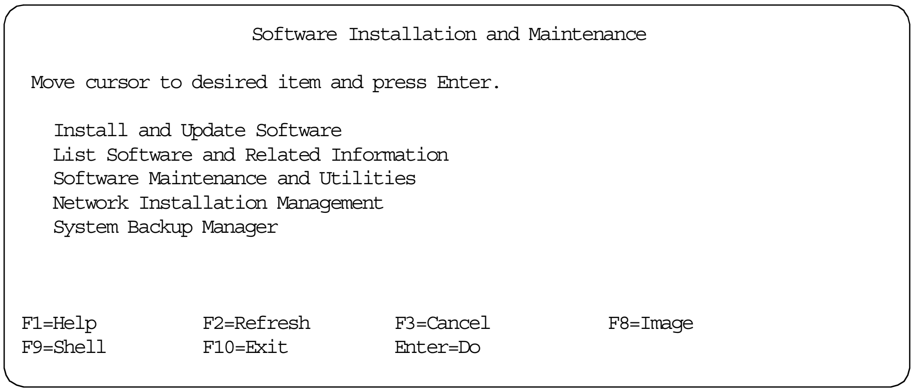
Note

> In the UNIX market, there are a lot of commercial applications that only use ASCII terminals making the ASCII (streams) user interface necessary. Also, many RS/6000 systems do not have graphical adapters and terminals for system management. Even on a graphical user interface, such as CDE, ASCII-based SMIT has the advantage of being faster (user interface part); so, a majority of system administrators are using SMIT in this way.

SMIT allows the use of fastpaths. This is a way to jump directly to a given SMIT menu. For example, the `smit tcpip` fastpath goes directly to the TCP/IP configuration menu.

Here are some other examples of SMIT fastpaths:

- To get directly to the installation screens, you can enter `smitty install`. This will display the below screen.

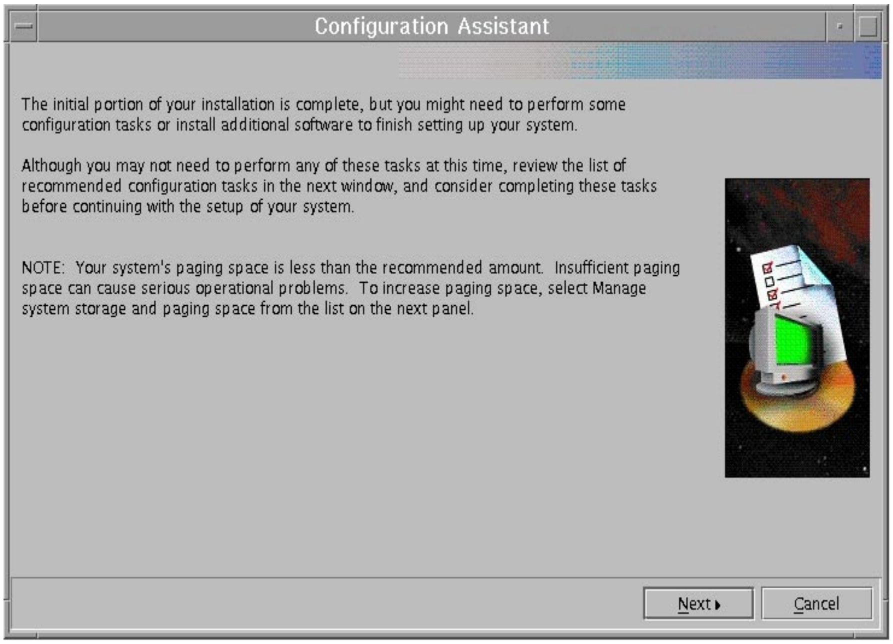


```
                      Software Installation and Maintenance

 Move cursor to desired item and press Enter.

   Install and Update Software
   List Software and Related Information
   Software Maintenance and Utilities
   Network Installation Management
   System Backup Manager



F1=Help          F2=Refresh        F3=Cancel          F8=Image
F9=Shell         F10=Exit          Enter=Do
```

- To get directly to the update software screen, you can enter `smitty update.`
- To get directly to the update ALL software screen, you can enter `smitty update_all.`

Configuration Assistant screen

The Configuration Assistant screen (see Figure 64 on page 214) is what a system administrator would see immediately after doing an overwrite or new installation. Its purpose is to simplify the initial configuration of a newly-installed system after the BOS installation. This interface is a simple straightforward interface whether it is run from an ASCII or graphics console.



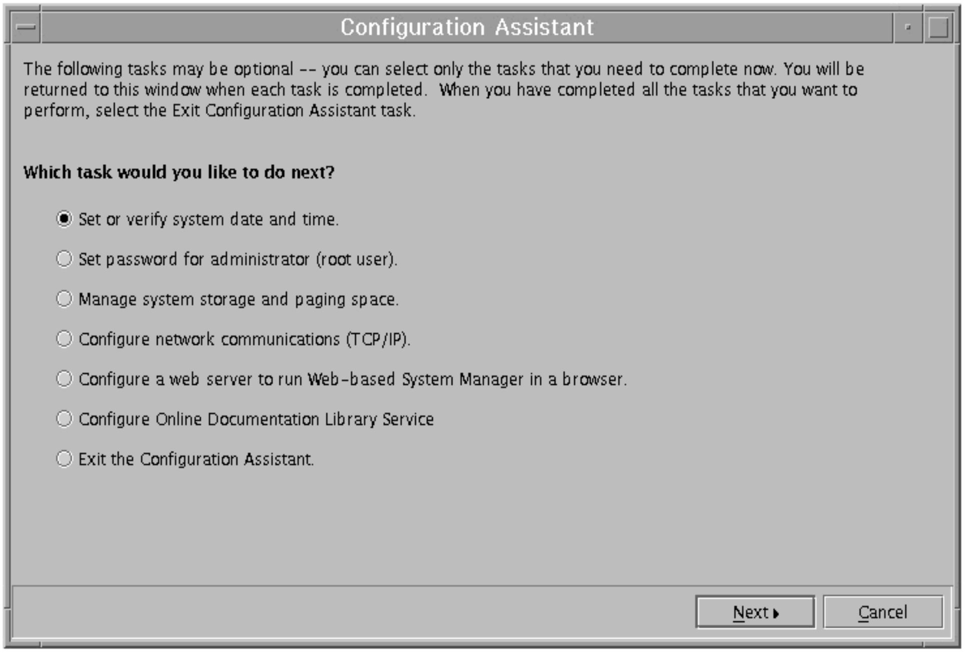
The initial portion of your installation is complete, but you might need to perform some configuration tasks or install additional software to finish setting up your system.

Although you may not need to perform any of these tasks at this time, review the list of recommended configuration tasks in the next window, and consider completing these tasks before continuing with the setup of your system.

NOTE: Your system's paging space is less than the recommended amount. Insufficient paging space can cause serious operational problems. To increase paging space, select Manage system storage and paging space from the list on the next panel.

Figure 64. Configuration Assistant first screen

If the system administrator clicks the Next button, he or she will see the menu with basic tasks that he or she should perform first to configure the system (see Figure 65 on page 215).



Configuration Assistant

The following tasks may be optional -- you can select only the tasks that you need to complete now. You will be returned to this window when each task is completed. When you have completed all the tasks that you want to perform, select the Exit Configuration Assistant task.

Which task would you like to do next?

● Set or verify system date and time.

○ Set password for administrator (root user).

○ Manage system storage and paging space.

○ Configure network communications (TCP/IP).

○ Configure a web server to run Web-based System Manager in a browser.

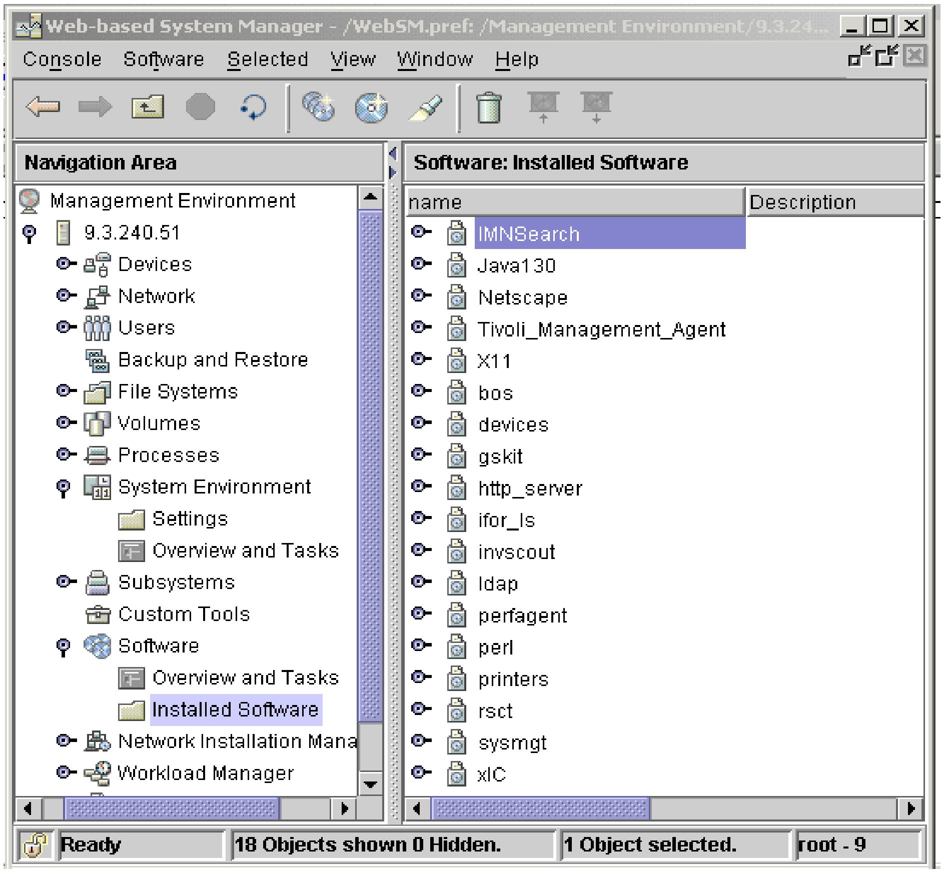
○ Configure Online Documentation Library Service

○ Exit the Configuration Assistant.

Figure 65. Configuration Assistant main menu

Web-based System Manager

Web-based System Manager is an application with which a system administrator can manage his or her system locally or over the WWW. The part of this application with which we can install additional SW or update the one installed is shown in Figure 66 on page 216.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.24...
Console
Software
Selected
View
Window
Help
Navigation Area
Software: Installed Software
Ready

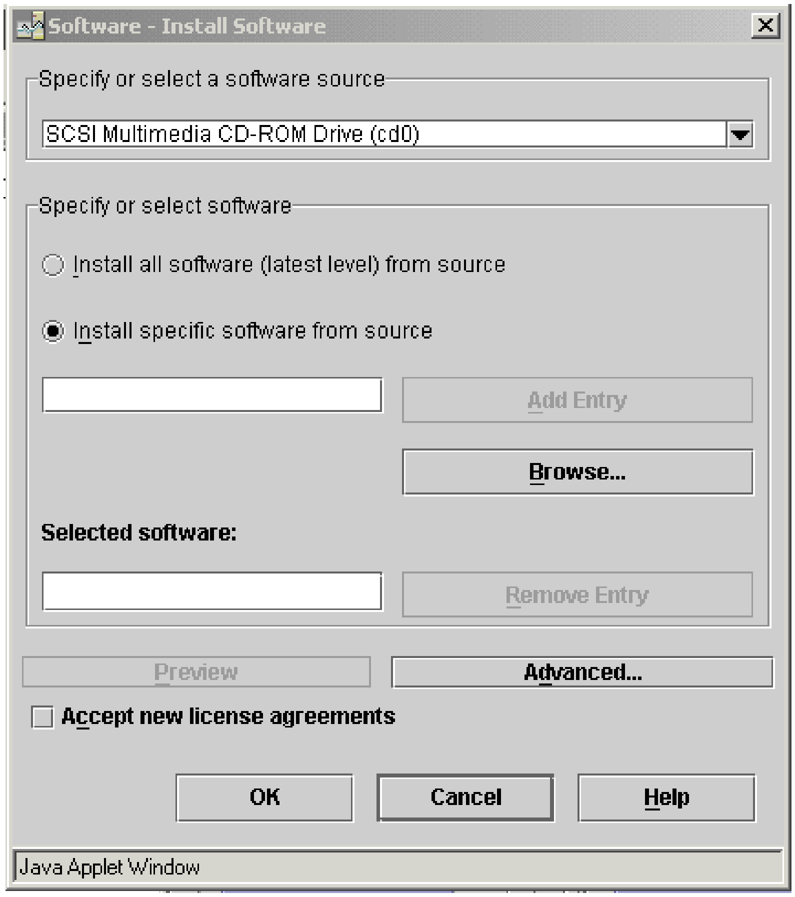
Management Environment
9.3.240.51
Devices
Network
Users
Backup and Restore
File Systems
Volumes
Processes
System Environment
Settings
Overview and Tasks
Subsystems
Custom Tools
Software
Overview and Tasks
Installed Software
Network Installation Mana
Workload Manager

name	Description
IMNSearch	
Java130	
Netscape	
Tivoli_Management_Agent	
X11	
bos	
devices	
gskit	
http_server	
ifor_ls	
invscout	
ldap	
perfagent	
perl	
printers	
rsct	
sysmgt	
xlC	

18 Objects shown 0 Hidden. 1 Object selected. root - 9

Figure 66. Web-based System Manager - Software Install

If the system administrator chooses **Software** -> **New Software** -> **Install Additional Software** -> **Advanced Method**, he or she can choose the software from installation media and install it (see Figure 67 on page 217).



Specify or select a software source: SCSI Multimedia CD-ROM Drive (cd0)

○ Install all software (latest level) from source

● Install specific software from source

□ Accept new license agreements

Figure 67. Web-based System Manager - Additional software installation

Network Installation manager (NIM)

NIM enables a system administrator to easily manage software installation and software updates in his or her network (LAN). In NIM configuration, you can group the systems you wish to install and that have the same or similar requirements, or you can tailor the installation for each separate system. For a picture of typical NIM topology, see Figure 68 on page 218.

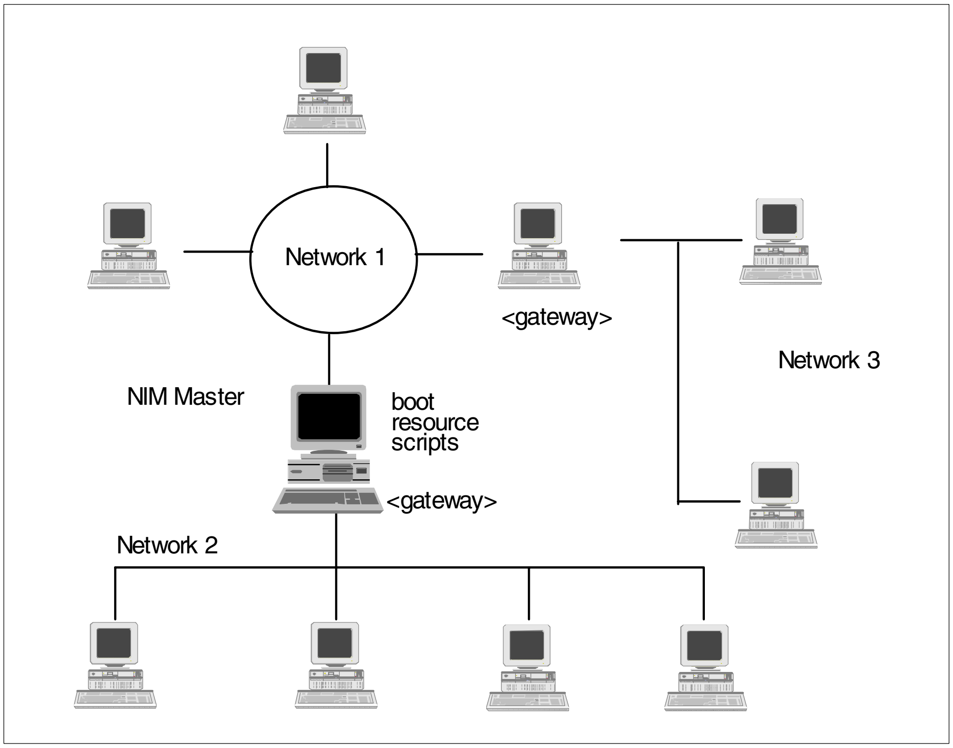


Figure 68. NIM in a typical LAN topology

More than one machine can be installed at the same time. The number of machines you can install simultaneously depends on the throughput of your network, the disk access throughput of the installation servers, and the platform type of your servers.

The NIM environment is comprised of client and server machines. A server provides resources, such as files and programs required for installation, to another machine. On the other hand, a machine that is dependent on a server to provide resources is known as a client.

The machines to be managed in the NIM environment, their resources, and the networks through which the machines communicate are all represented as objects within a central database that resides on the master server. Each of these objects has attributes that give it a unique identity, such as the network address of a machine or the location of a file or directory. With this information, you can install the base operating system and optional software on multiple machines managed from a central location.

NIM manages three types of clients:

- **Stand-alone** - This is a typical AIX machine. It has all of its own resources including the AIX operating system and user data resident on hard disk.
- **Diskless** - Diskless systems do not have a disk drive. A diskless client system can only boot from the network or from a boot master machine that supplies its operating system.
- **Dataless** - Dataless systems have a local disk drive, but they cannot boot from it. These clients rely on a master machine in much the same way as a diskless client. Usually, the paging space is installed on the local hard drive to avoid paging over the network.

NIM is also called *Bidirectional*. The NIM client can pull software from the NIM master. However, the NIM master can also be programmed to push software to clients throughout the network without any physical intervention on the client.

The push method can be scheduled and tailored for individual groups. This is an excellent way to upgrade software on multiple machines automatically at any time, day or night. These features make NIM a very powerful help tool for a system administrator in a large computer environment.

8.1.1.4 AIX 5L bundles

Generally, bundles are mechanisms used to easily install a predetermined set of software. A bundle is a list of software components (AIX License Program Products (LPPs)) specified in a flat ASCII file. System administrators can create bundles to use for systematic installation of multiple hosts/clients. The Web-based System Manager interface for software installation makes good use of software bundles.

Bundles on AIX 5L Installation media are:

Client

A collection of software products for single user systems running in a stand-alone or networked-client environment.

Server

A collection of software products, such as NFS Server, Print Server, and so on, for multiuser systems running in a stand-alone or networked environment.

Personal Productivity:

A collection of software products for graphical desktop systems running AIX and PC applications.

Application Development:
This bundle is the same as the client, with the addition of the development tools and utilities such as a linker, debugger and so on.

8.1.1.5 Language Support During Installation

When installing AIX, it is possible to determine which language will be used during the installation (in which language the installation menus are displayed) and also which language will be installed on the system.

Command `installp` deals with multiple language support, especially during first-time installation. In the case of a new installation, only language file sets of the selected language are installed for each software product. Additional language support can be added later from the install media. In the case of an upgrade or a migration, the install process will automatically update all applicable languages that are already on the system.

Note

All AIX-supported languages are provided with the AIX product media. There is no need to purchase a specific package of AIX to get the desired language(s).

8.1.1.6 Apply, Commit, Reject, Remove Software

Since AIX installs filesets as entities, it can also remove them as such. In the course of a normal installation, software is applied. Often, it is committed at the same time, especially in the case where the software is a first-time installation.

When applying software, AIX will move or preserve any existing version of the same software it finds on the system. This allows for the rejection of the software and the return of the system to its original state, that is, configured as it was before applying the software. Once an administrator is satisfied with the newly-applied software, he or she will commit the software. In essence, this removes the preserved copy of the software, thus, freeing disk space.

Software can also be totally removed from the system. Each software package from IBM contains an inventory list stored in ODM. This list is used to determine which files to remove from the system. In general, a correctly packaged product can be completely removed from a system. As a minimum, information regarding the software package is removed from the Vital Product Database (VPD).

All of these functions can be managed through the command line interface, through the SMIT interface, and through NIM. Applying and updating software can also be accomplished with the Web-based System Manager.

8.1.1.7 Installation on alternate disk

In AIX Version 4.3 and the later Version, the system administrator has the ability to install AIX on additional physical disk, apart from the one already installed. The new rootvg will be created (but is not active), and the system will be able to boot from this alternate disk if so desired. This functionality is very convenient when testing new versions of operating systems, updates, or other additional software products. It reduces the downtime to a considerable extent and avoids the need to restore system backup in case the new installation is rejected.

The Alternate Disk installation can be done in two ways:

Alternate mksysb disk installation

Installing a mksysb, which is a system backup image, requires a 4.3 or later mksysb image or 4.3 or later mksysb tape. The `alt_disk_install` command is called, specifying a disk or disks that are installed in the system but are not currently in use. The mksysb is restored to those disks so that, if the user chooses, the next reboot will boot the system on a new AIX system.

> **Note**
>
> If needed, the `bootlist` command can be run after the new disk has been booted, and the bootlist can be changed to boot back to the older version of AIX.

Cloning

Cloning allows the user to create a backup copy of the root volume group. Once created, the copy may be used as a backup, or it can be modified by installing additional updates. One possible use might be to clone a running production system and then install updates to bring the cloned rootvg to a later maintenance level. This would update the cloned rootvg while the system was still in production. Rebooting from the new rootvg would then bring the level of the running system up to the newly-installed maintenance level. If there was a problem with this level, simply changing the bootlist back to the original disk and rebooting would bring the system back to the old level.

Currently, you can run the `alt_disk_install` command on AIX 4.1.4 and higher systems for both of these functions. The bos.alt_disk_install.rte fileset must be installed on the system to do cloning to an alternate disk, and the bos.alt_disk_install.boot_images fileset must be installed to allow a mksysb install to an alternate disk.

The mksysb image that is used must be created before installation and must include all the necessary device and kernel support required for the system on which it is installed. No new device or kernel support can be installed before the system is rebooted from the newly-installed disk.

Note

The level of mksysb that you are installing must match the level of the bos.alt_disk_install.boot_images file set. At this time, AIX 5L, 4.3.3, 4.3.2, 4.3.1, and 4.3.0 mksysb images are supported. AIX 4.3.1 boot images are available only on the 4.3.1 installation media.

When cloning the rootvg volume group, a new boot image is created with the bosboot command. When installing a mksysb image, a boot image for the level of mksysb and platform type is copied to the boot logical volume for the new alternate rootvg. When the system is rebooted, the bosboot command is run in the early stage of boot, and the system will be rebooted again. This is to synchronize the boot image with the mksysb that was just restored. The system will then boot in normal mode.

At the end of the installation, a volume group, altinst_rootvg, is left on the target disks in the varied off state as a place holder. If varied on, it will show as owning no logical volumes, but it does, in fact, contain logical volumes. Their definitions have been removed from the ODM because their names now conflict with the names of the logical volumes on the running system. It is recommended that you do not vary on the altinst_rootvg volume group but just leave the definition there as a place holder.

When the system reboots from the new disk, the former rootvg will not show up in an lspv listing. The disks that were occupied by the rootvg will show up as not having a volume group. However, you can still use the bootlist command to change the bootlist to reboot from the old rootvg if necessary.

When the system is rebooted from the new altinst_rootvg, lspv will show the old rootvg as old_rootvg; so, you will know which disk or disks your previous rootvg was on. There is also a -q option in alt_disk_install that will allow you to query to see which disk has the boot logical volume so you can set your bootlist correctly for cases when old_rootvg has more than one disk.

8.1.2 AIX boot process

An RS/6000 system installed with AIX can be booted from:

- Hard Disk
- Tape
- CD-ROM
- Network

During boot up, the system tests the hardware, loads and executes the operating system, and configures devices. The AIX boot process is a multiple-phase process that provides the initial operating environment required to support the automatic configuration scheme for base devices. The base devices for a particular configuration of hardware are those required to boot the complete set of root file systems, disks, tapes, console devices, and other devices found in the system. The boot process is divided into three major phases Figure 69 on page 224.

There is a special type of boot called a *Service Mode Boot*. If the system administrator wants to use it, he or she must boot the system from the installation CD-ROM or system backup tape (or CD-ROM). Then, he or she chooses the **Start Maintenance Mode for System Recovery** option (see Figure 69 on page 224). This option is used for low-level system maintenance and problem determination.

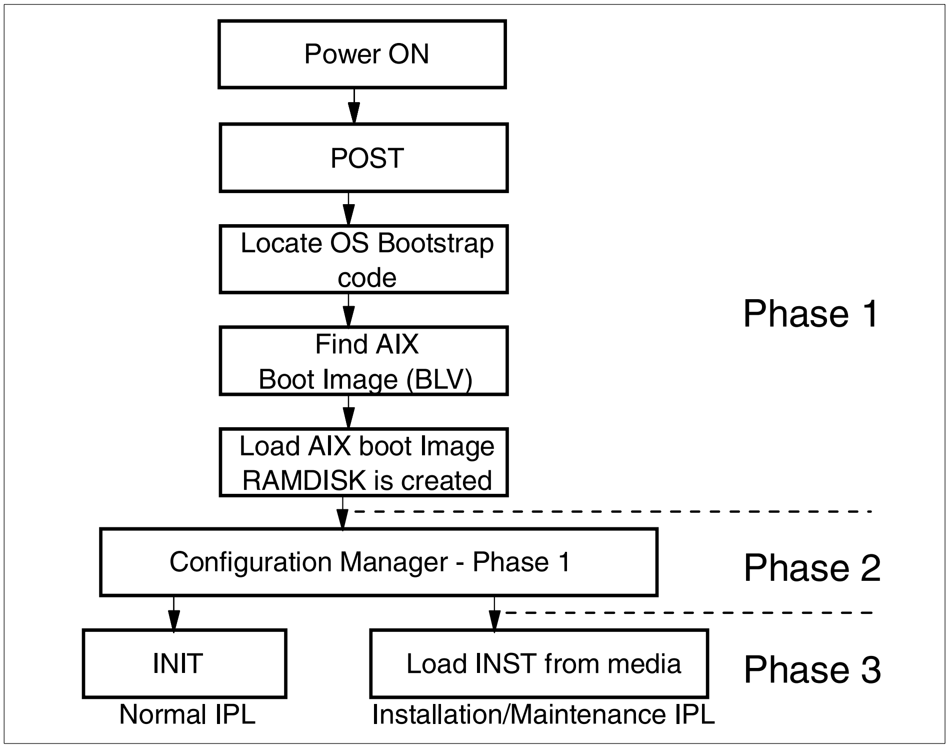

Power ON
POST
Locate OS Bootstrap code
Find AIX Boot Image (BLV)
Load AIX boot Image RAMDISK is created
Phase 1
Configuration Manager - Phase 1
Phase 2
INIT
Load INST from media
Phase 3
Normal IPL
Installation/Maintenance IPL

Figure 69. AIX boot process

The boot process differs slightly on different RS/6000 systems, but the major steps for all of them are listed below.

The first boot phase is called the ROS kernel initialization and consists of the following steps:

1. The On-Chip Sequencer (OCS) bring-up microprocessor (BUMP) checks to see if there are any problems with the system planar. Control is then passed to the Read-Only Storage (ROS), which performs Power-On Self Tests (POST). The ROS contains firmware stored in an EPROM.
2. The ROS initial program load (IPL) checks the user boot list, which is a list of bootable devices in a specific order. This list is contained in the Non-Volatile RAM (NVRAM). If this boot list is empty or points to an invalid boot device, the ROS firmware uses a default bootlist. The first valid boot device found in the boot list is used for system startup.

3. The first record or Program Sector Number (PSN) is checked on the boot device. If it is a valid boot record, it is read into memory and added to the initial program load (IPL) control block in memory.
4. The boot image is read sequentially from the boot device into memory starting at the location specified in the boot record. The disk boot image consists of the kernel, a RAM file system, and base-customized device information. The RAM file system is part of the boot image, is totally memory-resident, and contains all programs that allow the boot process to continue. The files in the RAM file system determine the type of boot. The `init` command on the RAM file system used during boot is actually the simple shell (ssh) program. The ssh program controls the boot process by calling the rc.boot script.
5. Control is passed to the kernel, which begins system initialization.
6. Process 1 executes init, which executes phase 1 of the rc.boot script. The first step for rc.boot is to determine from which device the machine was booted. The boot device determines which devices should be configured on the RAM file system. If the machine is booted over the network, the network devices need to be configured so that the client's file systems can be remotely mounted. In the case of a tape or CD-ROM boot, the console is configured to display the BOS installation menus. After the `rc.boot` script finds the boot device, the appropriate configuration routines are called from the RAM file system.

The steps of the second phase, base device configuration, are described below (see Figure 70 on page 226):

1. The boot script calls the restbase program to build the customized Object Database Manager (ODM) database in the RAM file system from the compressed customized data.
2. The boot script starts the configuration manager, which accesses phase 1 configuration rules to configure the base devices.
3. The configuration manager starts the sys, bus, disk, SCSI, and the Logical Volume Manager (LVM) and rootvg volume group configuration methods.
4. The configuration methods load the device drivers, create special files, and update the customized data in the ODM database.

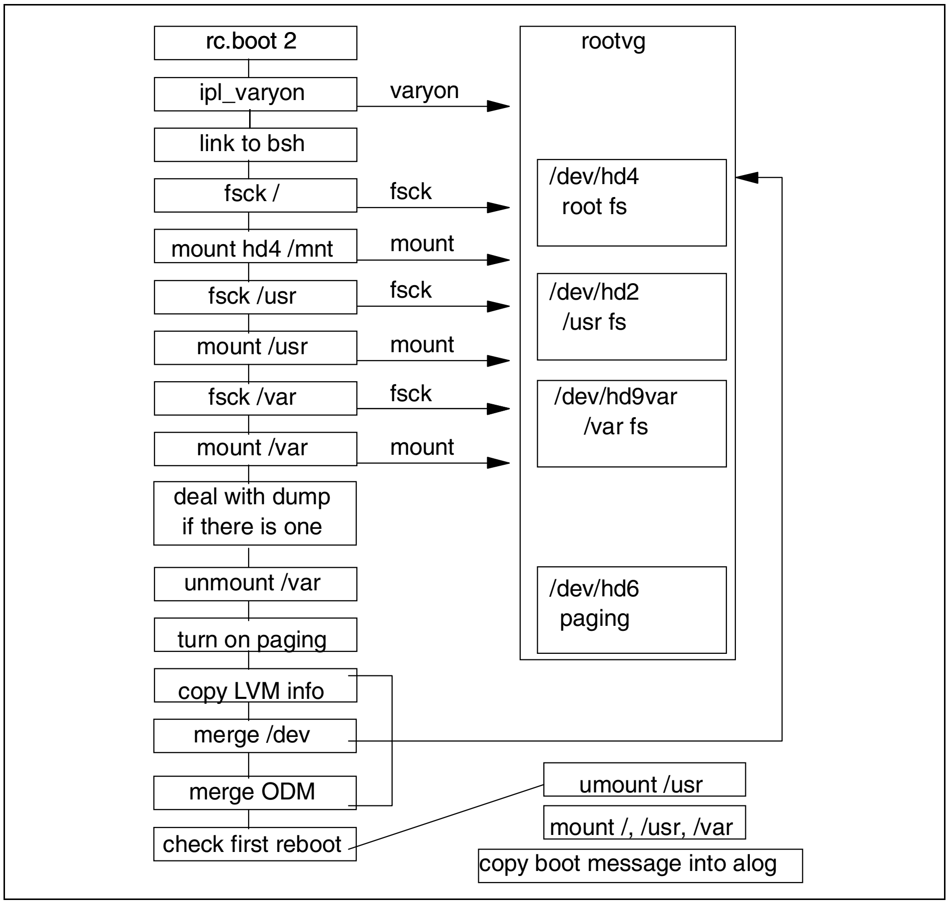


Figure 70. AIX boot process - Phase 2

The steps of the third phase, system boot, are described below (see Figure 71 on page 228):

1. The init process starts phase 2 execution of the rc.boot script. Phase 2 of rc.boot includes the following steps:

   a. Call the ipl_varyon program to vary on the rootvg volume group.

   b. Mount the hard disk file systems onto the RAM file system.

   c. Run swapon to start paging.

   d. Copy the customized data from the ODM database in the RAM file system to the ODM database in the hard disk file system.

   e. Unmount temporary mounts of hard disk file systems and then perform permanent mounts of / (root), /usr, and /var file systems.

   f. Exit the rc.boot script.

2. After phase 2 of rc.boot, the boot process switches from the RAM file system to the hard disk root file system.

3. Then, the init process executes the processes defined by records in the /etc/inittab file. One of the instructions in the /etc/inittab file executes phase 3 of the rc.boot script, which includes the following steps:

   a. Mount the /tmp hard disk file system.

   b. Start configuration manager phase 2 to configure all remaining devices.

   c. Use the `savebase` command to save the customized data to the boot logical volume.

   d. Exit the rc.boot script.

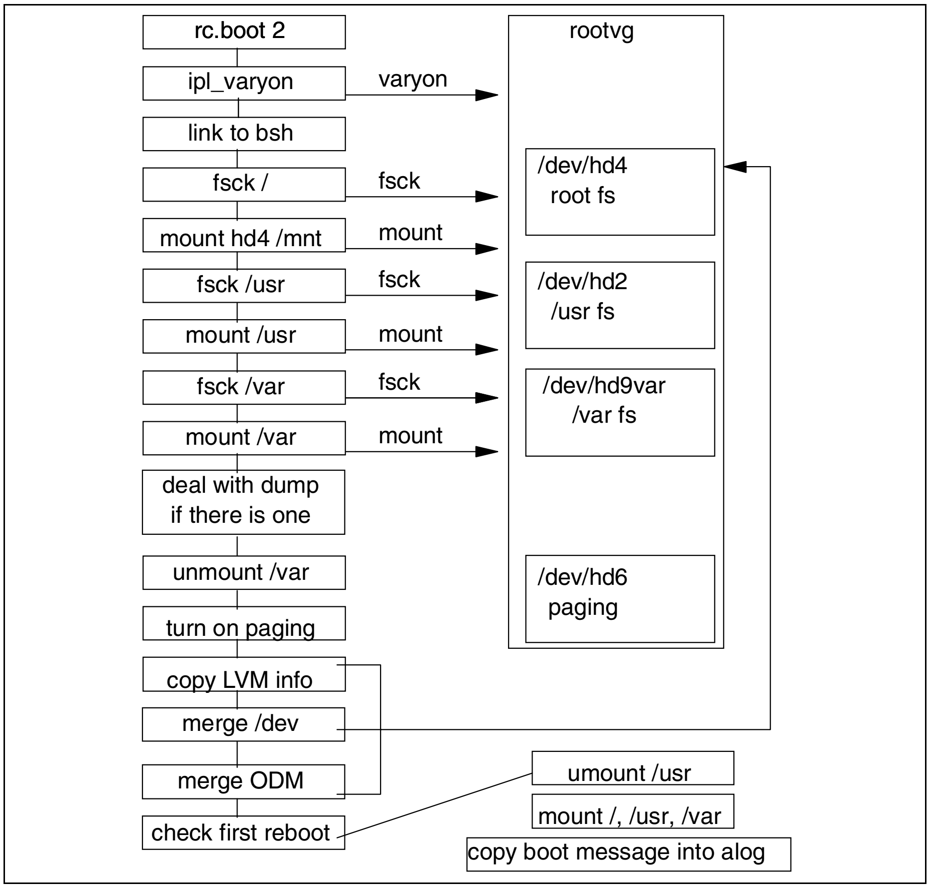


Figure 71. AIX boot process - Phase 3

After this phase, the system is up and running and ready for use.

8.1.3 AIX configuration management

Configuration management is the way system information can be stored and retrieved by the system. AIX uses the Object Database Manager (ODM) database. In this section, we will look at how devices are automatically detected, configured by the system, and stored on the system. The following section will describe configuration management in an AIX environment.

8.1.3.1 AIX Object Data Manager (ODM)

The Object Data Manager (ODM) is an object-oriented database that contains AIX system data. ODM provides a more robust, secure, shareable resource than what previously existed in the UNIX environment (ASCII stanza and colon files). Information is stored and maintained as objects with associated characteristics. The ODM can also be used to manage data for application programs. Access to information stored in the ODM and the ability to add, delete, or change information can be accomplished through the SMIT interface, the command line, or C language subroutines.

The basic components of ODM are object classes and objects. An object class is a group of objects with the same definition. The object class is conceptually similar to an array of structures, with each object being a structure that is an element of the array.

An object, a member of a defined object class, is an entity that requires storage and management of data. The object is the equivalent of the record in the object class. This record is further divided in fields in which a particular attribute is stored.

System data managed by the ODM includes:

- Device configuration information
- Display information for SMIT (menus, selectors, and dialogs)
- Vital Product Database (VPD) for installation and update procedures
- Communications configuration information
- System resource information

System data not managed by the ODM includes the following (for compatibility reasons with other UNIX systems):

- Information about file systems (stored in /etc/filesystems)
- Information about printer queues (stored in /etc/qconfig file)
- Information about users and groups (stored in /etc and /etc/security directories)

System configuration

A primary function of the ODM is to manage the system resource information or system configuration. The system configuration information is divided into three categories:

- **Predefined information** - This database contains configuration information related to all devices that can be installed on the system. On AIX Version 4, the user can make the Predefined Database contain information about devices not present on the system by choosing to install additional device support. This database is comprised of the following object classes:

  - Predefined Devices Object Class (PdDv)
  - Predefined Connections Class (PdCn)
  - Predefined Attributes Object Class (PdAt)

- **Customized Information** - This database contains configuration information related about actual devices installed (configured) on the system. This database is comprised of the following object classes:
  - Customized Devices Object Class (CuDv)
  - Customized Dependency Class (CuDep)
  - Customized Attribute Object Class (CuAt)
  - Customized Device Driver Class (CuDvDr)
  - Customized Vital Product Data Object Class (CuVPD)
- **Boot Information** - This boot ODM contains the bare minimum predefined and customized databases required to perform a system boot. Thus, only devices considered necessary for boot are included in this object class.

Note

All predefined object classes (files) are stored in the /usr/lib/objrepos directory. By default, all customized object classes (files) are stored in the /etc/objrepos directory. If the system administrator wants to keep them somewhere else, he or she could change the value of the system variable, ODMDIR. For example:

```
export ODMDIR=/etc/objrepos_modified
```

ODM commands

In general, system administrator would not have to deal directly with the ODM database. Most interactions with the ODM are handled through SMIT sessions or by AIX device management commands. However, the ODM database can be manipulated directly from the command line with high-level commands. These commands can also be used by applications that run on the AIX operating system.

Some of the most often used commands are:

- `odmshow` - Displays an object class definition on the screen.
- `odmget` - Retrieves objects from an object class in stanza format. This command accepts wildcards for SQL-like queries.

- `odmadd` - This command is used to add a new object to an object class.
- `odmcreate` - Creates object classes required for applications that will use the ODM database. Produces the .c and .h files necessary for ODM application development.
- `odmchange` - This command changes all objects within an object class that meets specified criteria.
- `odmdelete` - This command deletes all objects that meet a specific criteria from the object class. If no criteria is specified, all objects in that object class are deleted.
- `odmdrop` - This will remove an entire object class.

These and other functions are available as an API subroutine as defined in /usr/lib/libodm.a file.

8.1.3.2 AIX Device Configuration Management

In AIX, we can work with devices via three different user interfaces: Command line, SMIT, and Web-based System Manager.

AIX command line device configuration

Often, an experienced administrator will perform device configuration and management from the AIX command line. This allows for automation of repetitive tasks. Often, complete device configuration can be performed with just one AIX command. The `cfgmgr` command detects all devices installed on the system and configures them based upon information contained in the ODM. This command is also run by the system during the boot process.

Many of the command line commands are quite simple to use but also allow for a great deal of flexibility by way of command line flags:

- **List** - These commands, in the form of lsxxx, are used to list currently defined or configured devices, for example, `lsdev`, `lsattr`, and `lscfg`.
- **Make** - These commands, in the form of mkxxx, are used to create or define a new device, for example, `mkdev` and `mknotify`.
- **Change** - These commands, in the form of chxxx, change or modify an existing device, for example, `chdev` and `chdisp`.
- **Remove** - These commands, in the form of rmxxx, remove previously defined devices from the ODM, for example, `rmdev` and `rmserver`.
- **Define** - These commands, in the form of defxxx, define new devices to the ODM, for example, `defif`, and `definet`.

- **Undefine** - These commands, in the form of undefxxx, undefine devices in ODM but leave the configuration intact, for example, `undefif` and `undefinet`.
- **Configure** - These commands, in the form of cfgxxx, configure devices that have previously been defined, for example, `cfgmgr` and `cfginet`.
- **Unconfigure** - These commands, in the form of ucfgxxx, unconfigure previously-defined devices, for example, usfgif and ucfginet.

Here are some examples:

- Configuring a 4mm tape device connected to SCSI adapter scsi0, at SCSI address 3, with a block size of 512 bytes:

```
# mkdev -c tape -t'4mm4gb' -s'scsi' -p'scsi0' -w '3,0' -a
block_size='512'
```

- Changing the SCSI address of SCSI adapter scsi0 to 5:

```
# chdev -l 'scsi0' -a id='5'
```

- Removing the CD-ROM device:

```
# rmdev -l cd0 -d
```

- Listing all disks configured on your system:

```
# lsdev -C -c disk
```

8.1.4 AIX system administrator interface

In this section, we will cover different user interfaces available for system administrators to perform common system administration tasks, such as storage management, user management, device management, network management and so on.

Traditionally, in UNIX operating systems, system administration tasks were done by UNIX commands and by editing so-called stanza files (plain ASCII files). This was and still is a very powerful method, but it requires a knowledge about many system commands with a very large number of parameters and careful editing of plain ASCII files. To help system administrators perform their tasks, there are two powerful tools available on AIX: Web-based System Manager and SMIT (System Management Interface Tool).

8.1.4.1 Web-based System Manager

Web-based System Manager enables a system administrator to manage an AIX system either locally from a graphics terminal or remotely from a PC or RS/6000 client. Information is entered through the use of GUI components on the client side. The information is then sent over the network to the

Web-based System Manager server, which runs the commands necessary to perform the required action.

Web-based System Manager is implemented using the Java programming language. The implementation of Web-based System Manager in Java provides:

- **Cross-platform portability** - Any client platform with a Java 1.1-enabled Web browser is able to run a Web-based System Manager client object.
- **Remote administration** - A Web-based System Manager client is able to administer an AIX machine remotely through the Internet.
- **A richer and more flexible GUI environment** than is available with either HTML forms or Java Script.

Web-based System Manager is a family of applications for local and remote administration of AIX. For using Web-based System Manager, following packages are to be installed on your AIX system, as shown in Table 16

Table 16. Web-based System Manager filesets

Package Names	Description
sysmgt.help.Lang.websm	Web-based System Manager Extended Helps
sysmgt.help.msg.Lang	Web-based System Manager Context Sensitive Helps
sysmgt.sguide	Web-based System Manager TaskGuide Runtime Environment
sysmgt.websm	Web-based System Manager Applications

Web-based System Manager is started with the `wsm` command on AIX system (see Figure 72 on page 234).

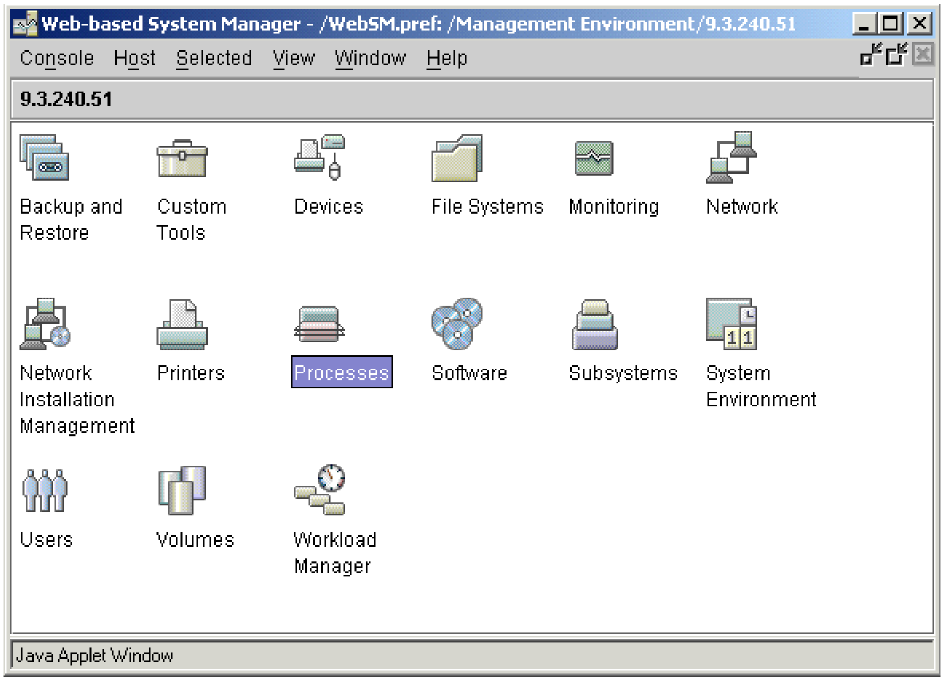
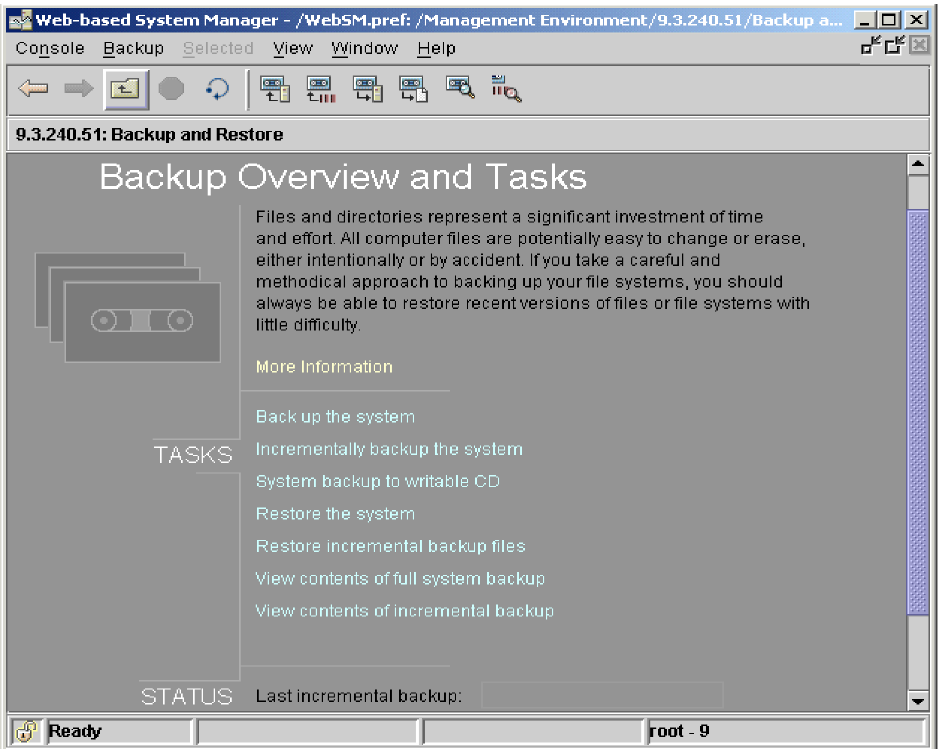


Figure 72. Web-based System Manager

A system administrator can navigate through Web-based System Manager and administer his or her system in the following areas:

Backup

The system can be backed up on tape drive or on CD-ROM drive. By clicking on the CD-R icon, the system administrator is guided through all the necessary steps in order to create a bootable backup of the system (`rootvg`) to the CD_ROM. See Figure 73 on page 235.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51/Backup a...
Console
Backup
Selected
View
Window
Help
9.3.240.51: Backup and Restore
Ready
root - 9

Backup Overview and Tasks

Files and directories represent a significant investment of time and effort. All computer files are potentially easy to change or erase, either intentionally or by accident. If you take a careful and methodical approach to backing up your file systems, you should always be able to restore recent versions of files or file systems with little difficulty.

More Information

TASKS

- Back up the system
- Incrementally backup the system
- System backup to writable CD
- Restore the system
- Restore incremental backup files
- View contents of full system backup
- View contents of incremental backup

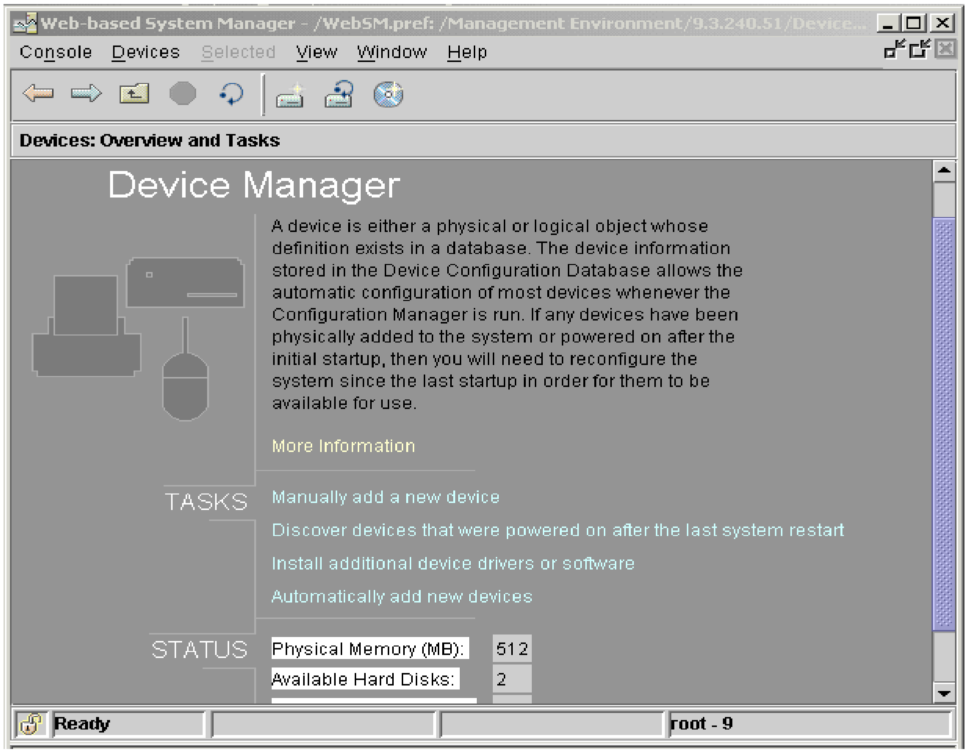
STATUS

Last incremental backup:

Figure 73. Web-based System Manager - Backups

Devices

This window shows the list of the devices configured on the system (See Figure 74 on page 236). By highlighting the specific device, the system administrator can unconfigure (put into a *Defined* state), configure (put into an *Available* state), delete, or change the properties of a device.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51/Device...
Console
Devices
Selected
View
Window
Help
Devices: Overview and Tasks
Ready
root - 9

Device Manager

A device is either a physical or logical object whose definition exists in a database. The device information stored in the Device Configuration Database allows the automatic configuration of most devices whenever the Configuration Manager is run. If any devices have been physically added to the system or powered on after the initial startup, then you will need to reconfigure the system since the last startup in order for them to be available for use.

More Information

TASKS

- Manually add a new device
- Discover devices that were powered on after the last system restart
- Install additional device drivers or software
- Automatically add new devices

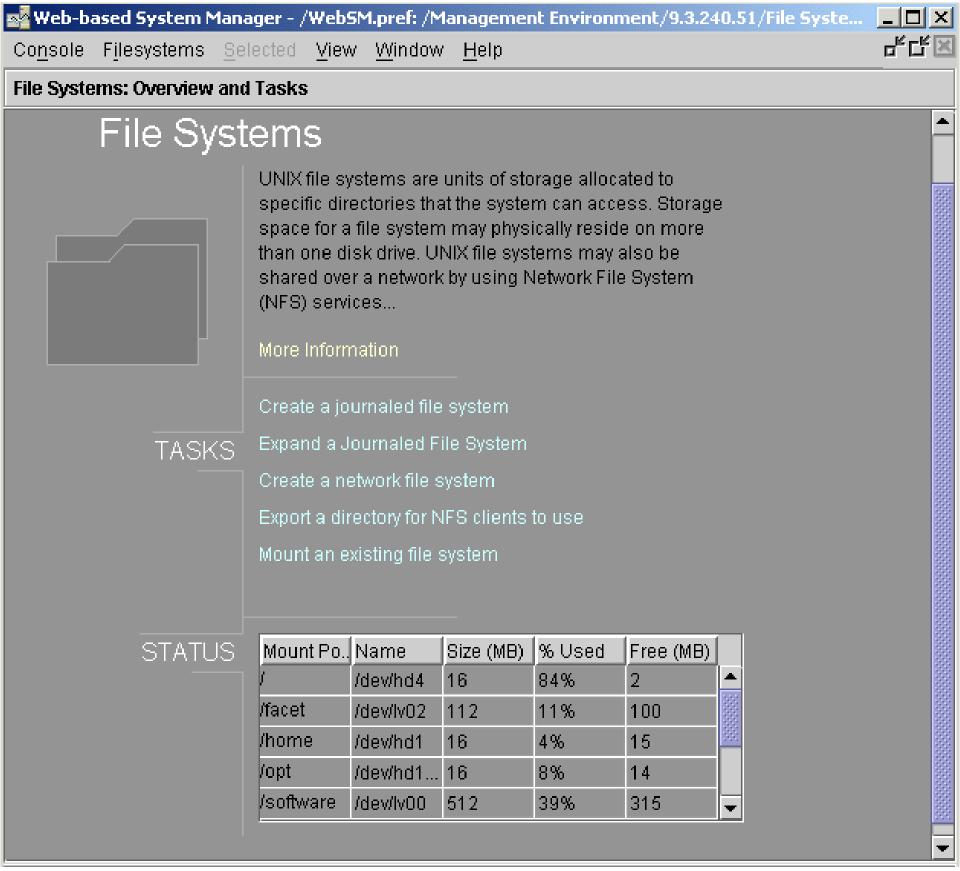
STATUS

Physical Memory (MB):	512
Available Hard Disks:	2

Figure 74. Web-based System Manager - Devices

File systems

A system administrator can create new file systems (JFS, CD-ROM, NFS, Cached), mount/unmount file systems, back up file systems, restore file systems from backup, change the properties of a file system, such as size, mount options, and read-write or read-only permissions, and defragment file systems. Figure 75 on page 237 shows the File Systems screen of the Web-based System Manager.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51/File Syste...
Console
Filesystems
Selected
View
Window
Help
File Systems: Overview and Tasks
File Systems
TASKS
STATUS

UNIX file systems are units of storage allocated to specific directories that the system can access. Storage space for a file system may physically reside on more than one disk drive. UNIX file systems may also be shared over a network by using Network File System (NFS) services...

More Information

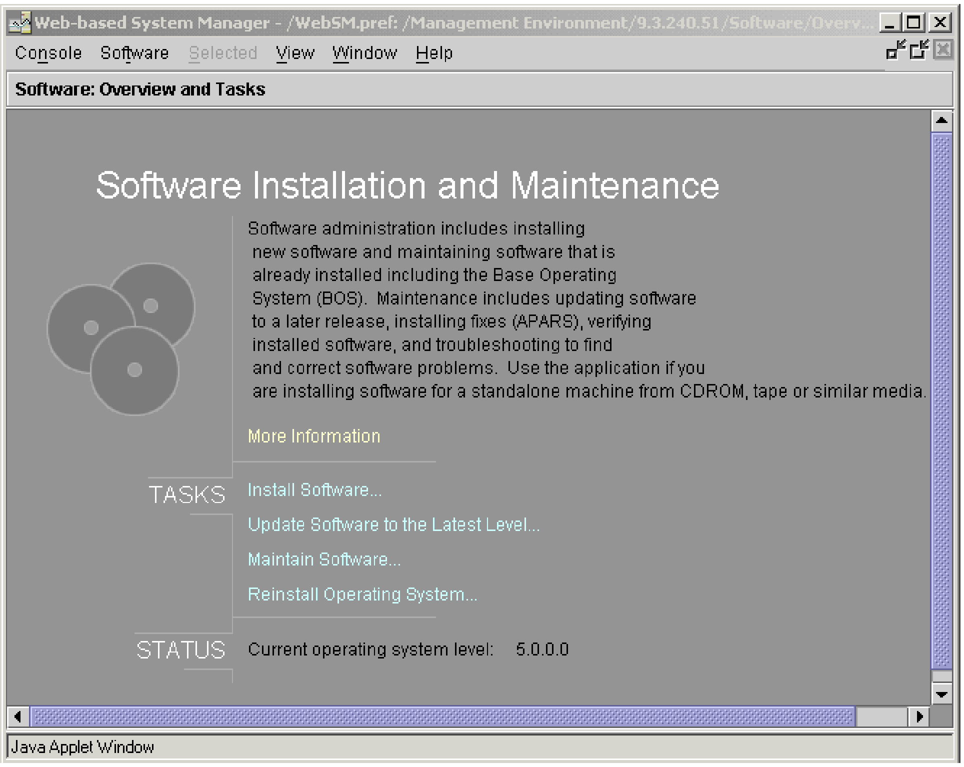
- Create a journaled file system
- Expand a Journaled File System
- Create a network file system
- Export a directory for NFS clients to use
- Mount an existing file system

Mount Po...	Name	Size (MB)	% Used	Free (MB)
/	/dev/hd4	16	84%	2
/facet	/dev/lv02	112	11%	100
/home	/dev/hd1	16	4%	15
/opt	/dev/hd1...	16	8%	14
/software	/dev/lv00	512	39%	315

Figure 75. Web-based System Manager - File systems

Software installation and maintenance

The system administrator can list all of the installed SW, install new software products (LPPs), and commit, reject, or verify installed SW. He or she can also manage the NIM client (for more information about NIM, see UNRESOLVED Network Installation manager (NIM)). Figure 76 on page 238 shows the Software Installation and Maintenance screen of the Web-based System Manager.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51/Software/Overv...
Console Software Selected View Window Help
TASKS
STATUS
Java Applet Window

Software: Overview and Tasks

Software Installation and Maintenance

Software administration includes installing new software and maintaining software that is already installed including the Base Operating System (BOS). Maintenance includes updating software to a later release, installing fixes (APARS), verifying installed software, and troubleshooting to find and correct software problems. Use the application if you are installing software for a standalone machine from CDROM, tape or similar media.

More Information

Install Software...

Update Software to the Latest Level...

Maintain Software...

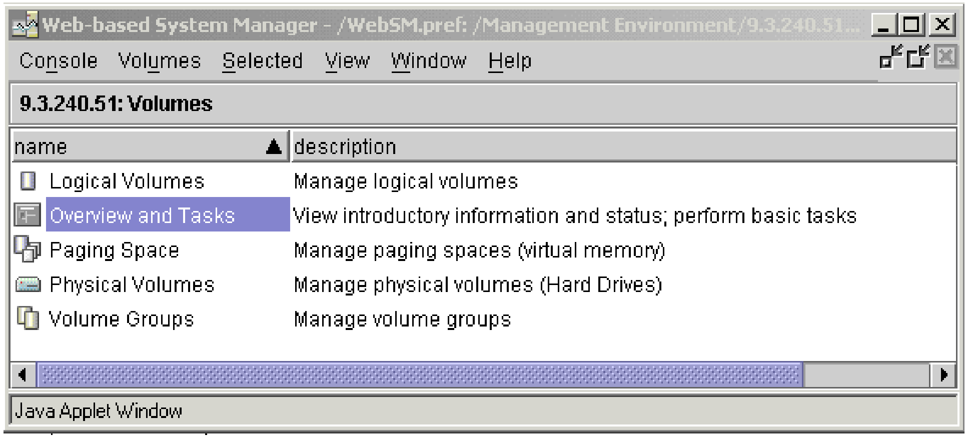
Reinstall Operating System...

Current operating system level: 5.0.0.0

Figure 76. Web-based System Manager - Software installation & maintenance

Storage management

In this pane (see Figure 77 on page 239), the system administrator can define, configure, and manage volume groups, logical volumes, and paging space. For more details about storage management in AIX, see section 6.1, “AIX storage management” on page 105.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51...
Console Volumes Selected View Window Help
Java Applet Window

9.3.240.51: Volumes

name	description
Logical Volumes	Manage logical volumes
Overview and Tasks	View introductory information and status; perform basic tasks
Paging Space	Manage paging spaces (virtual memory)
Physical Volumes	Manage physical volumes (Hard Drives)
Volume Groups	Manage volume groups

Figure 77. Web-based System Manager - Storage management

Network management

A system administrator can configure and manage the network environment from this pane. All defined network protocols (if installed), such as TCP/IP, SNA or X.25, and NFS and NIS, can be defined and configured from here. For more details about Network management, see Chapter 10, “Networking” on page 361. The Network Management screen is shown in Figure 78.

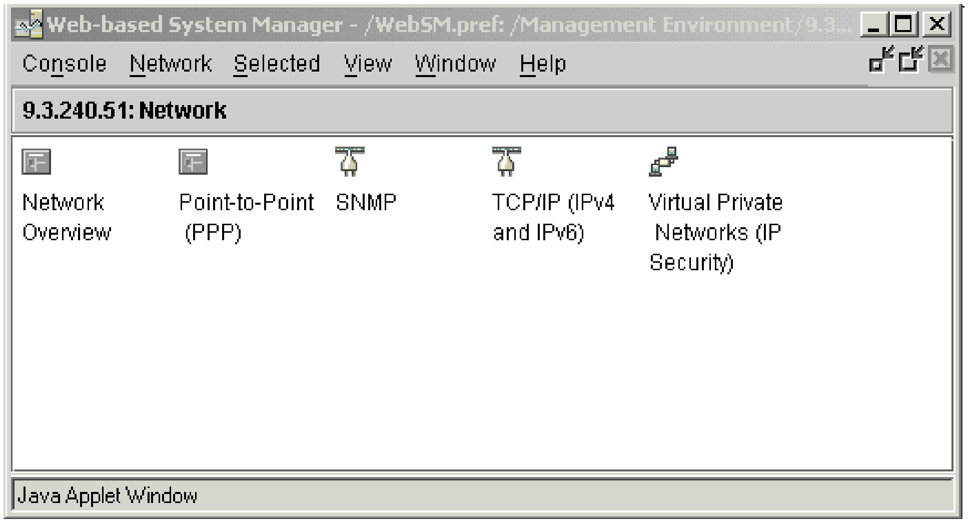
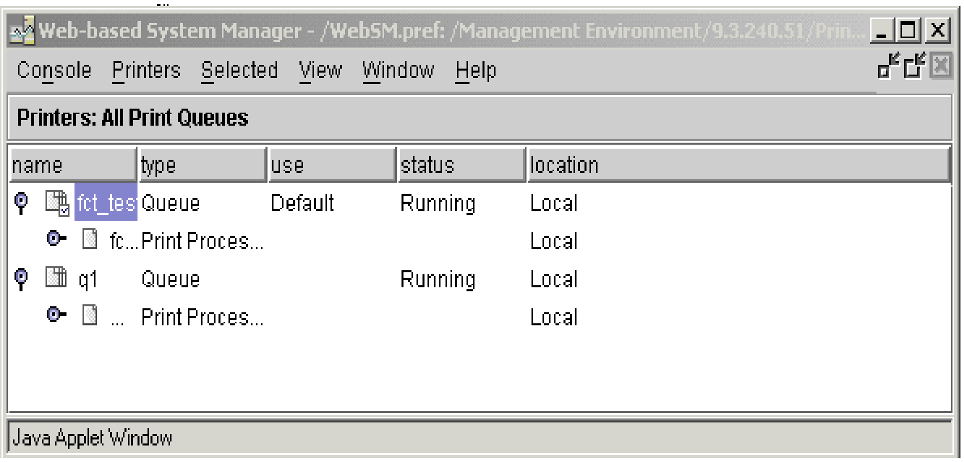

Web-based System Manager - /WebSM.pref: /Management Environment/9.3...
Console Network Selected View Window Help
9.3.240.51: Network
Network Overview
Point-to-Point (PPP)
SNMP
TCP/IP (IPv4 and IPv6)
Virtual Private Networks (IP Security)
Java Applet Window

Figure 78. Web-based System Manager - Network management

Printing management

The system administrator can install support for new printers, define and configure printers, and manage a print spooling system from this panel. He or she can also change the properties of existing devices and queues. See Figure 79 on page 240.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51/Prin...
Console
Printers
Selected
View
Window
Help
Java Applet Window

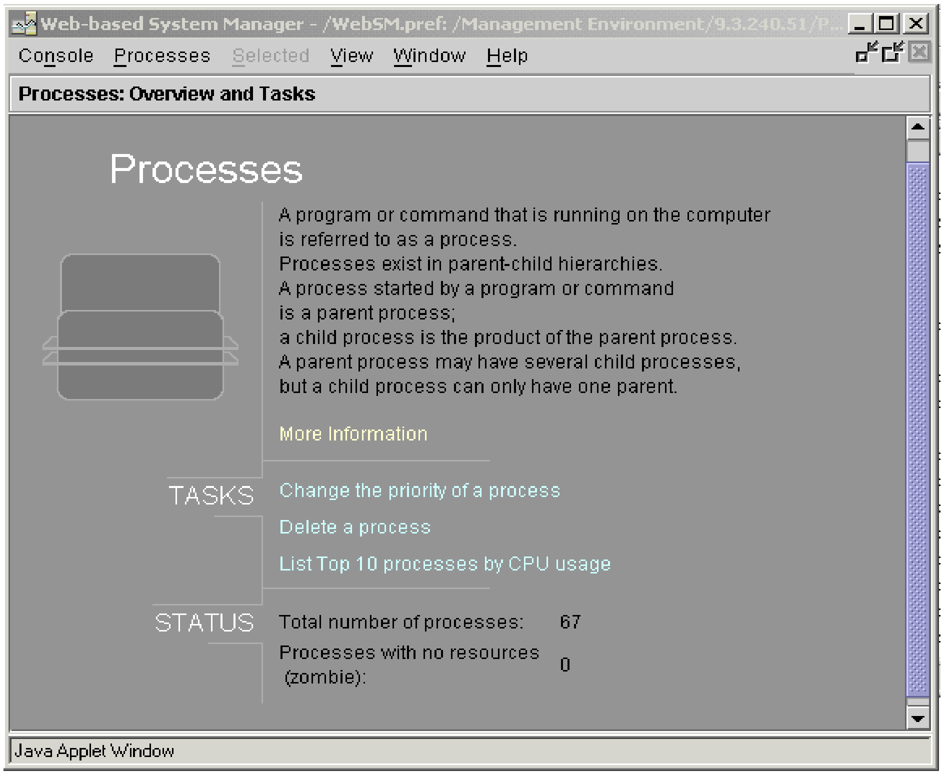
Printers: All Print Queues

name	type	use	status	location
fct_tes	Queue	Default	Running	Local
fc...	Print Proces...			Local
q1	Queue		Running	Local
...	Print Proces...			Local

Figure 79. Web-based System Manager - Printing management

Processes management

The system administrator can list all existing processes on the system and sort them by their names, ids, or use of system resources. He or she can also stop (kill) the processes or re-prioritize them. See Figure 80 on page 241.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51/P...
Console
Processes
Selected
View
Window
Help

Processes: Overview and Tasks

Processes

A program or command that is running on the computer is referred to as a process.
Processes exist in parent-child hierarchies.
A process started by a program or command is a parent process;
a child process is the product of the parent process.
A parent process may have several child processes, but a child process can only have one parent.

More Information

TASKS

Change the priority of a process

Delete a process

List Top 10 processes by CPU usage

STATUS

Total number of processes: 67

Processes with no resources (zombie): 0

Java Applet Window

Figure 80. Web-based System Manager - Processes

System Environment Management

The system administrator can define different system consoles, change the date and time of the system, change characteristics of the operating system, such as the maximum number of processes allowed per user, maintaining I/O history, and so on, change the default user interface (CDE or command line user interface), manage system dump (location, start), manage the licenses of the operating system, broadcast messages to users, configure the Internet environment, manage the language environment, and stop (shut down) the system. See Figure 81 on page 242.

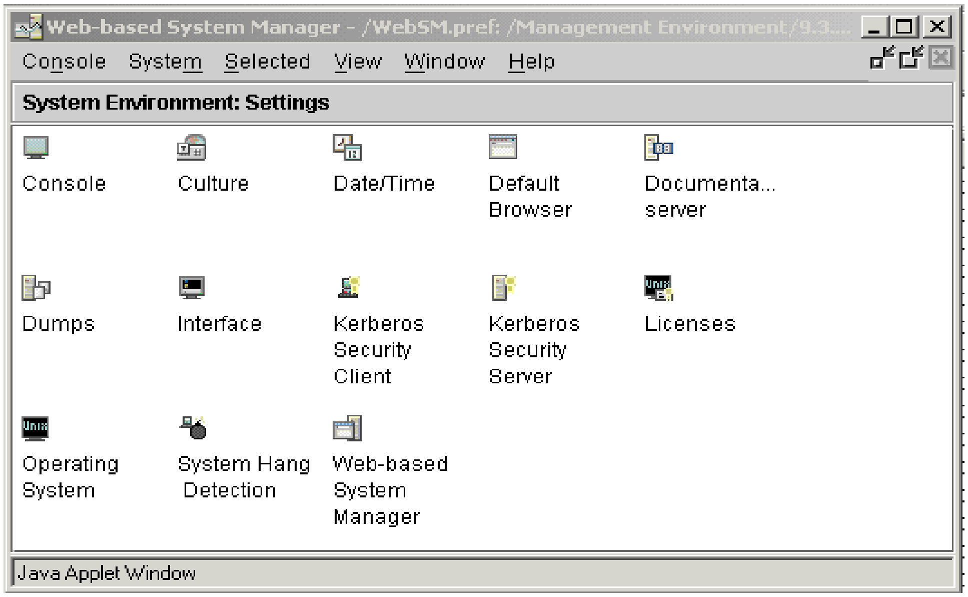
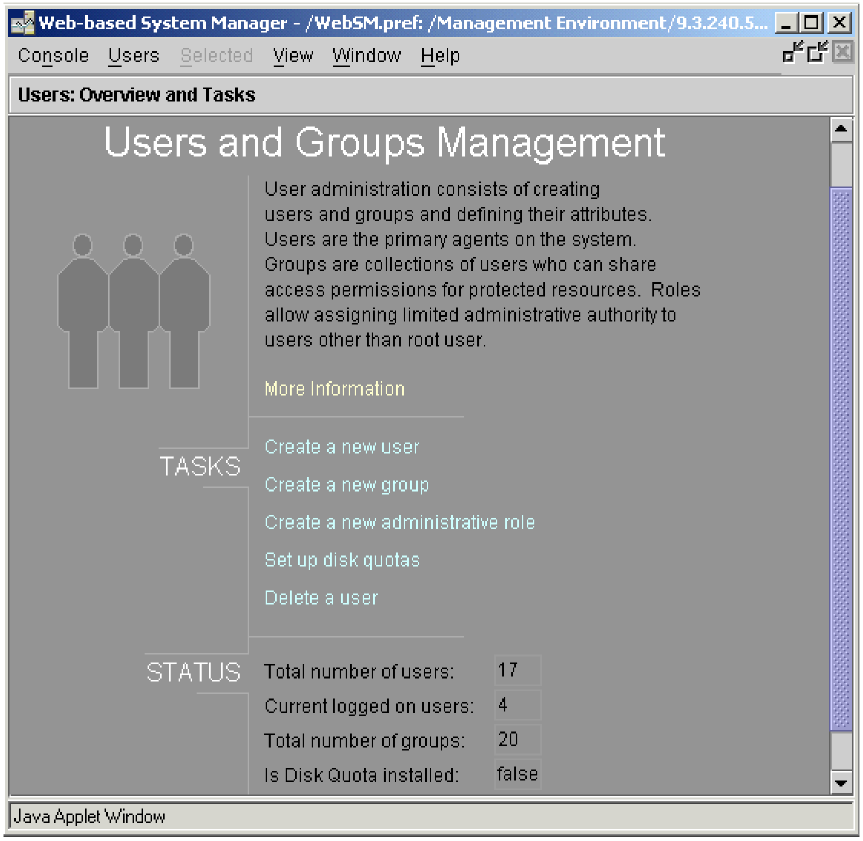


Figure 81. Web-based System Manager - System Environment Management

Users Management

In this pane, the system administrator can define users and groups and manage the existing ones, for example, adding the user to another group. Disk quotas are also managed from here. See Figure 82 on page 243.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.5...
Console Users Selected View Window Help
Users: Overview and Tasks
Java Applet Window

Users and Groups Management

User administration consists of creating users and groups and defining their attributes. Users are the primary agents on the system. Groups are collections of users who can share access permissions for protected resources. Roles allow assigning limited administrative authority to users other than root user.

More Information

TASKS

- Create a new user
- Create a new group
- Create a new administrative role
- Set up disk quotas
- Delete a user

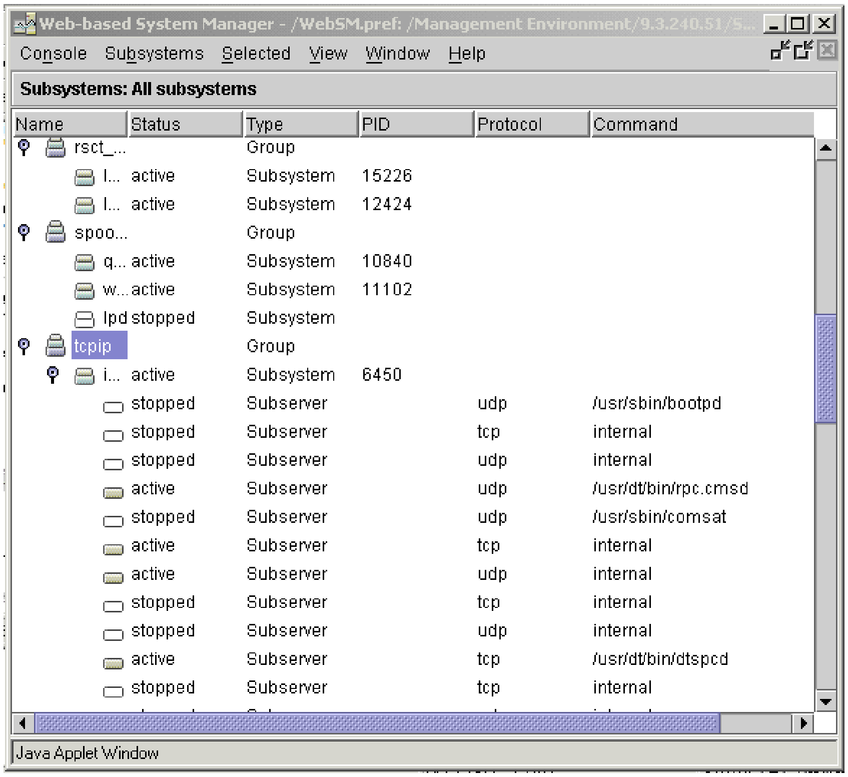
STATUS

Total number of users:	17
Current logged on users:	4
Total number of groups:	20
Is Disk Quota installed:	false

Figure 82. Web-based System Manager - User Management

Subsystems Management

AIX subsystems, such as iforls (licensing), inetd (TCP/IP), and spooler (printing), are managed from here. They can be started or stopped and traced for problem determination. See Figure 83 on page 244.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51/S...
Console
Subsystems
Selected
View
Window
Help
Subsystems: All subsystems
Java Applet Window

Name	Status	Type	PID	Protocol	Command
rsct_...		Group			
l...	active	Subsystem	15226		
l...	active	Subsystem	12424		
spoo...		Group			
q...	active	Subsystem	10840		
w...	active	Subsystem	11102		
lpd	stopped	Subsystem			
tcpip		Group			
i...	active	Subsystem	6450		
	stopped	Subserver		udp	/usr/sbin/bootpd
	stopped	Subserver		tcp	internal
	stopped	Subserver		udp	internal
	active	Subserver		udp	/usr/dt/bin/rpc.cmsd
	stopped	Subserver		udp	/usr/sbin/comsat
	active	Subserver		tcp	internal
	active	Subserver		udp	internal
	stopped	Subserver		tcp	internal
	stopped	Subserver		udp	internal
	active	Subserver		tcp	/usr/dt/bin/dtspcd
	stopped	Subserver		tcp	internal

Figure 83. Web-based System Manager - Subsystems Management

Note

Web-based System Manager allows remote administration sessions to be carried out using the Secure Socket Layer (SSL) protocol. This allows all data transmitted on the network between the Web-based System Manager client and the system being managed to be encrypted and, thereby, prevent unauthorized systems from viewing the data.

8.1.4.2 SMIT

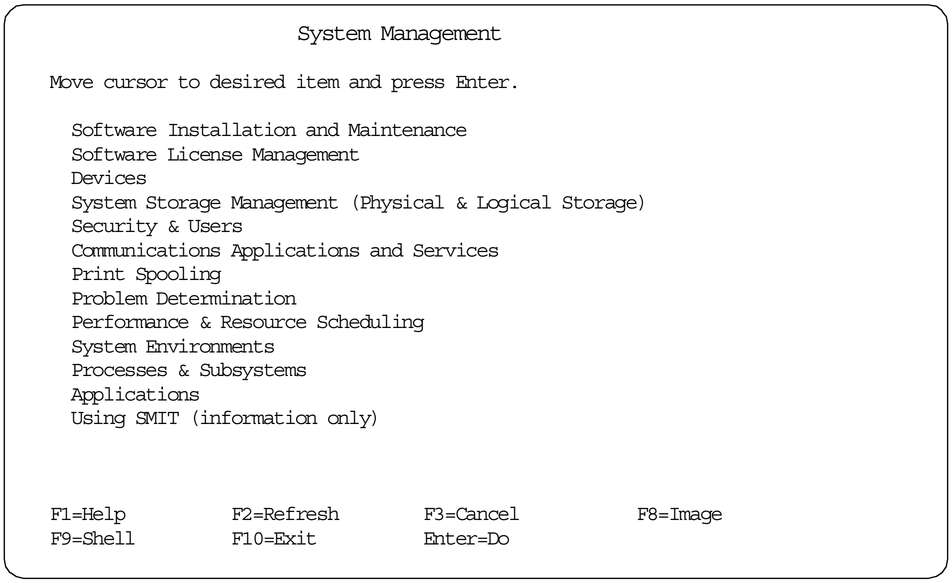
The System Management Interface Tool (SMIT) is an interactive interface designed to simplify system management tasks. The `smit` command displays a hierarchy of menus that can lead to interactive dialogues. SMIT builds and runs commands as directed by the user. Because SMIT runs commands, you need the appropriate authority to execute the commands that SMIT runs.

The `smit` command takes you to the top level of the menu hierarchy. To directly enter into a lower menu level, a fastpath parameter can be used. The fastpath parameter assists you as you become familiar with the commands. For example, you can enter `smit chuser` to go directly to the dialog in which you can change the characteristics of the user.

SMIT creates two files, smit.script and smit.log, when it is executed. The smit.script file automatically records the commands with the command flags and parameters used. The smit.script file can be used as an executable shell script to repeat administration tasks on the same or other systems. This is extremely useful when dealing with a large number of systems. There is information in the smit.log file about the history of using the smit. All menus and commands that the system administrator issues are logged. These two files are growing without limitation and can become quiet large over the time; so, it is one of the responsibilities of the system administrator to monitor them and, eventually, archive or delete them.

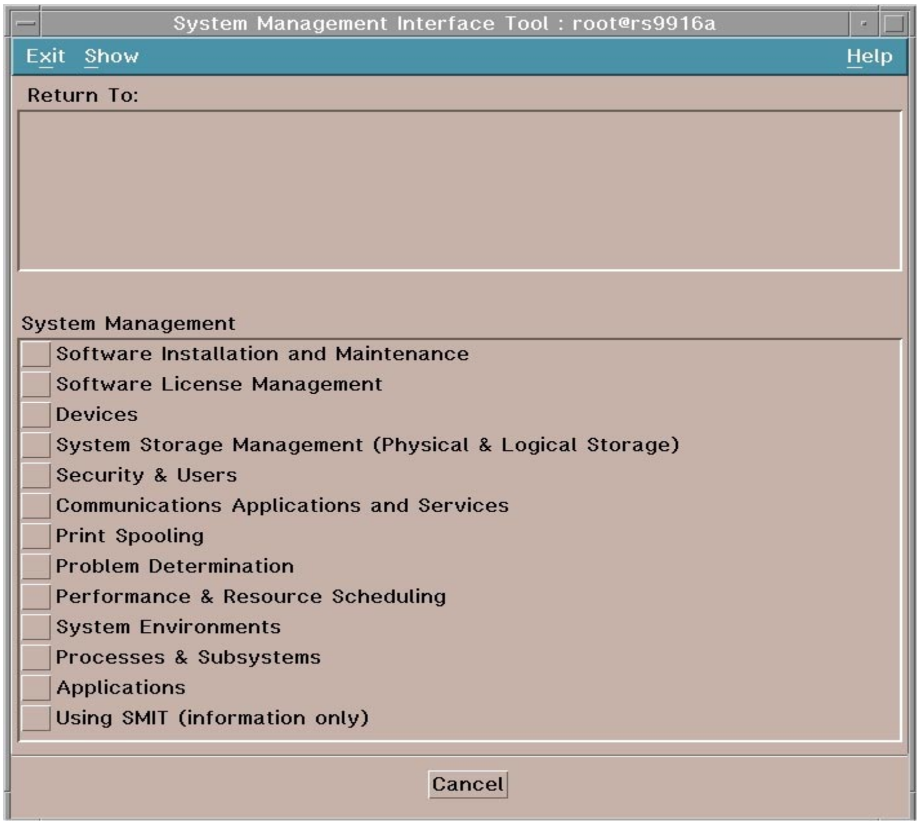
With SMIT, it is also possible to log a SMIT command without executing it by using the `-xs` option when starting `smit` (`smit -xs`). The command can then be read from the smit.script file and executed directly later or within a shell script.

SMIT provides two interfaces: An ASCII interface and a Motif based user interface. When you execute SMIT on an ASCII console or on the Low Function Terminal (LFT), the ASCII interface will appear as shown in below screen.



```
                       System Management

 Move cursor to desired item and press Enter.

   Software Installation and Maintenance
   Software License Management
   Devices
   System Storage Management (Physical & Logical Storage)
   Security & Users
   Communications Applications and Services
   Print Spooling
   Problem Determination
   Performance & Resource Scheduling
   System Environments
   Processes & Subsystems
   Applications
   Using SMIT (information only)



 F1=Help         F2=Refresh      F3=Cancel       F8=Image
 F9=Shell        F10=Exit        Enter=Do
```

If you issue the `smit` command in an X Window environment, the Motif SMIT user interface will be displayed as shown in Figure 84 on page 247. In this case, if the system administrator prefers to use the ASCII SMIT user interface, he or she must issue the `smit -C` command or the `smitty` command.


System Management Interface Tool : root@rs9916a

System Management

□ Software Installation and Maintenance
□ Software License Management
□ Devices
□ System Storage Management (Physical & Logical Storage)
□ Security & Users
□ Communications Applications and Services
□ Print Spooling
□ Problem Determination
□ Performance & Resource Scheduling
□ System Environments
□ Processes & Subsystems
□ Applications
□ Using SMIT (information only)

Figure 84. SMIT Motif-based user interface

8.1.5 AIX Backup/Restore

Backup and restore is one of the most important system administration tasks. In understanding the importance of system backup and recovery on AIX, one must keep four items in perspective:

- One backup method does not fit all needs.
- All data on a system may not need to be backed up.
- Being able to back up data from many systems is very important.
- The ability to restore that data is even more important.

AIX provides several tools to assist a system administrator with the task of backup and recovery. Even though AIX does provide some simple-to-use tools for a system backup, including a bootable backup, the administrator must design and implement a backup strategy to suit the needs of his or her IT environment.

Careful consideration has to be given to classifying data, data retention, and levels of protection. This is not to imply that backup and recovery are difficult to perform. On the contrary, it is rather simple to perform. The more difficult part is the coherent design, implementation, and management of the overall strategy.

To help the administrator perform this set of tasks, AIX provides many UNIX standard tools as well as a few tools unique to AIX. When properly used, these tools can provide a very strong level of enterprise-wide data protection.

8.1.5.1 AIX backup tools

AIX provides several tools that can be used for backup/recovery. However, most of these commands are contained in a fileset, bos.sysmgt.sysbr, that may not be installed on all systems. This fileset is shipped with all AIX packages, but an administrator may have chosen not to install it on all systems on the network:

- `tar` - This is one of the traditional UNIX command line-oriented backup commands. The `tar` command manipulates archives by writing files to (or retrieving files from) an archive storage medium, such as disk, tape, or other removable media. Although this command has some limitations, (ACLs cannot be backed up with tar!), it is still widely used for quick and portable backups.

Note

The tar command is not enabled for files larger than 2 GB due to limitations imposed by XPG/4 and POSIX.2 standards.

- cpio - The cpio command copies files into and out of archive storage and directories. This is another traditional UNIX command. It is a rather flexible tool that works well with special character devices as output devices. The cpio command can also work with input/output redirection and provides a good set of pattern-matching facilities. The cpio command is, generally, very portable between UNIX systems.

Note

The cpio command is not enabled for files greater than 2 GB in size due to limitations imposed by XPG/4 and POSIX.2 standards.

- dd - This is another very powerful and versatile traditional UNIX command. It is commonly used to convert and copy data to and from non-AIX systems and from one media type to another. dd does not group multiple files into one archive. It is used to manipulate and move raw data. The input and output block size can be specified to take advantage of raw devices.
- rdump - This is a network command that backs up files by file system onto a remote machine's device. Using the backup command format, the files are copied to a device on the remote machine. rdump can perform nine different levels of backups ranging from a *full complete* to selected *increments*. The /etc/dumpdates file is used to keep track of the last time the command was run and what level of backup was performed.
- pax - pax is a POSIX-compliant archive utility that can read and write tar and cpio archives. The pax command extracts and writes member files of archive files, writes lists of the member files of archives, and copies directory hierarchies.In AIX 5L, the pax command is enhanced to support a 64-bit POSIX-defined data format, which is used by default. The objective of this command is to allow archiving of large files, such as dumps. The commands cpio and tar do not support files used as input larger than 2 GB, because they are limited by their 32-bit formats.If you have to archive files larger than 2 GB, the only available option is the pax command. Suppose you have several tar archives with a total size exceeding the 2 GB limit. With the following command, you can create an archive for all of them:

```
# pax -x pax -wvf soft.pax ./soft?.tar
```

The default mode is for pax (without the -x option) to behave as tar. The -x option will allow pax the ability to work with files larger than 2 GB, a behavior tar does not have. This enhancement is also available on AIX Version 4.3.3 service releases.

- backup - This is the most useful command for performing backups in AIX. It is used as a part of the mksysb script, which performs System Backup on AIX and savevg script, which is used to back up other non-rootvg volume groups. It backs up files by name or by file system. This command is unique to AIX. backup can perform multiple levels of backups and has its own file-compression scheme. A nice feature of backup is its ability to allow the backup to span multiple volumes, such as tapes or CD-ROMs.
- mksysb - This is a shell script which creates a bootable system image of rootvg. The image can be created on a file, or, more commonly, on a tape or CD-ROM.This command is unique to AIX. It is a very powerful yet simple-to-use command. mksysb can utilize an exclude file to help filter which files to skip over or exclude particular files from the backup.
- savevg - This command is similar to mksysb, but is used for creating non-bootable images of volume groups other than rootvg.

The system administrator has several options for backing up all the files in his or her system. It can be done in one step using the tar or backup command, but, in this case, only the files themselves would be backed up. If the system has to be restored, the system administrator must first configure all volume groups, all logical volumes, and all file systems and only then restore the files from tape or CD-ROM. The more appropriate and convenient way is to use mksysb to back up the rootvg. It will store all information about this volume group, such as which logical volumes are configured, how large they are, which file systems are created, their mounting points, and so on. In the restore operation, the volume group will be recreated exactly as before, and all files will be restored. For other data, which resides outside rootvg, the savevg command is recommended.

8.1.5.2 On-line backup of mirrored logical volumes

In AIX Version 4.3 and higher, the system administrator can back up one copy of a mirrored logical volume while the other copy of logical volumes is still in use by applications and users. After backup, only the physical partitions that changed during backup must be synchronized.

8.1.5.3 On-line Backup of Journaled File System

In AIX Version 4.3.3 and higher, the system administrator can also make an on-line backup of a file system created on a mirrored logical volume in the same sense as described in section 8.1.5.2, “On-line backup of mirrored logical volumes” on page 250. For more details see section 6.1.5.12, “On-line Journaled File System backup” on page 130.

8.1.5.4 Other backup packages

IBM offers two other separately-orderable backup packages. These packages provide functionality beyond what is already supplied on the AIX operating system. Both of these packages are designed to work in a networked environment.

AIX System backup and recovery (Sysback/6000)

This product makes extensive use of the `mksysb` command as well as backing up other non-rootvg, volumes. It also contains some features for the restore part of the process that can be valuable to an administrator, such as changing the file system size and composition during the restore phase as well as restoring non-rootvg data. It even has features to assist with the network configuration.

Tivoli Storage Manger (TSM)

TSM is an IBM client/server product for enterprise storage management. It contains modules for backup/restore, archive/retrieve, hierarchical storage management and disaster recovery. This product is meant for enterprise-wide backup and recovery. It can handle data from a wide variety of clients and the server can reside on several IBM and non-IBM platforms as well. It is a central backup-oriented server. All backups go to the server for safe keeping and tape management.

8.1.6 AIX process management

In this section, we will describe how the processes (tasks or jobs) are managed in the AIX 5L.

When a user starts an application on AIX, he or she is actually starting one or more processes. Management of the processes can be done via the command line (the usual method), SMIT, or Web-based System Manager.

After starting a process, the user can do several things with it:

- List the processes

  List all the processes he or she is running (`jobs` or `ps` command)

- Stop the process

  If the user does not want the process to finish by itself or if he or she wants to interrupt the process for whatever reason, the user can send a special signal (interrupt) to the process, and the process will stop the execution. This is done with the `kill` command. The user can only stop his or her processes, but the system administrator can stop all the user’s processes and system processes.

- Put the process in the background

  If the user starts a process and later realizes that it will take a long time to finish it, and he or she wants to do some other tasks on their system, he or she can put the process in the background where it will execute until completion (with a slightly lower priority). This can be done with the `bg` command.

- Prioritize the process

  The process is always started with one initial priority for all users. If the user wants to prioritize his or her processes, he or she can decrease the priority of certain processes so that the others will receive more system resources and finish in a shorter time. The system administrator (root user) can also increase the priority of the processes above the initial level.The commands `nice` and `renice` are used for this task.

8.1.6.1 AIX Workload Manager (WLM)

The new concept of Workload Manager was created to better manage the system resources; it was first announced in AIX Version 4.3.3 and then enhanced in AIX 5L.

Workload Manager concept and terminology

WLM is designed to give the system administrator greater control over how the scheduler and Virtual Memory Manager (VMM) allocate CPU and physical memory resources to processes. It can be used to prevent different jobs from interfering with each other and to allocate resources based on the requirements of different groups of users.

The major use of WLM is for large SMP systems, and it is typically used for server consolidation, where workloads from many different server systems, (print, database, general user, transaction processing systems, and so on) are combined. These workloads often compete for resources and have differing goals and service level agreements. At the same time, WLM can be used in uniprocessor workstations to improve responsiveness of interactive work by reserving physical memory. WLM can also be used to manage individual SP nodes.

Following Table 17 on page 253 describes the WLM specific terminologies.

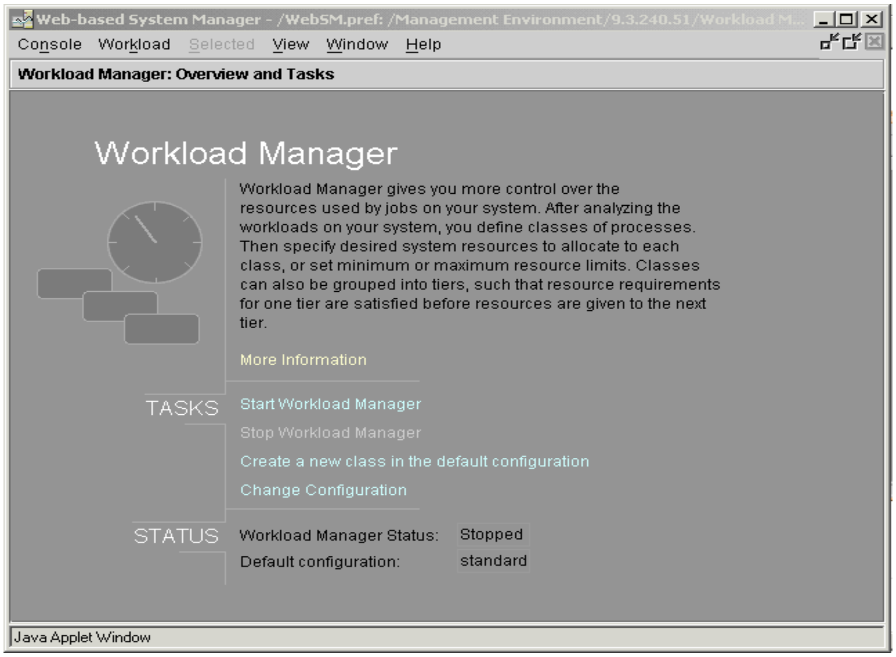
Table 17. WLM terminology

Terms	Description
class	A class is a collection of processes (jobs) that has a single set of resource limits applied to it. WLM assigns processes to the various classes and controls the allocation of system resources among the different classes. WLM allows system administrators to set up a hierarchy of classes with two levels by defining superclasses and subclasses. At the superclass level, the determination of resource entitlement is based on the total amount of each resource managed by WLM available on the machine. At the subclass level, the resource shares and limits are based on the amount of each resource allocated to the parent superclass.
classification mechanism	A set of class assignment rules that determines which processes are assigned to which classes (superclasses or subclasses within superclasses).
class assignment rule	A class assignment rule indicates which values within a set of process attributes result in a process being assigned to a particular class (superclass or subclass within a superclass).
process attribute value	A value that a process has for a process attribute. The process attributes can include attributes such as user ID, group ID, and application path name.
resource-limitation values	a set of values that WLM maintains for a set of resource utilization values. These limits are completely independent of the resource limits specified with the setrlimit subroutine.
resource target share	The shares of a resource that are available to a class (subclass or superclass). These shares are used with other class shares (subclass or superclass) at the same level and tier to determine the desired distribution of the resources between classes at that level and tier.
resource-utilization value	The amount of a resource that a process or set of processes is currently using in a system.
scope-of-resource collection	The level at which resource utilization is collected and the level at which resource-limitation values are applied.

Terms	Description
process class properties	The set of properties that are given to a process based on the classes (subclass and superclass) to which it is assigned.
class authorizations	a set of rules that indicates which users and groups are allowed to perform operations on a class or processes and threads in a class. This includes the authorization to manually assign processes to a class or to create subclasses of a superclass.
class tier	the position of the class within the hierarchy of resource limitation desirability for all classes. The resource limits (including the resource targets) for all classes in a tier are satisfied before any resource is provided to lower tier classes.

Workload Manager configuration

WLM configuration is performed through the preferred interface, the Web-based System Manager (Figure 85), through a text editor and AIX commands, or through the AIX administration tool SMIT.


Web-based System Manager - /WebSM.pref: /Management Environment/9.3.240.51/Workload M...
Console
Workload
Selected
View
Window
Help
Workload Manager: Overview and Tasks
Workload Manager
More Information
TASKS
Start Workload Manager
Stop Workload Manager
Create a new class in the default configuration
Change Configuration
STATUS
Java Applet Window

Workload Manager gives you more control over the resources used by jobs on your system. After analyzing the workloads on your system, you define classes of processes. Then specify desired system resources to allocate to each class, or set minimum or maximum resource limits. Classes can also be grouped into tiers, such that resource requirements for one tier are satisfied before resources are given to the next tier.

Workload Manager Status: Stopped

Default configuration: standard

Figure 85. WLM Overview and tasks menu on Web-based System Manager

Workload Manager is not started by default on the system, and the system administrator must start it manually (with the `startwlm` command) or add an entry in /etc/inittab file (this can be done via SMIT or Web-based System Manager).

All information about classes, limits, shares, and so on is in the /etc/wlm directory in flat ASCII files in stanza format. We can have multiple sets of configurations and start Workload Management each time with a different configuration, which is very useful in testing periods.

Note

For more information on previous Workload Manager architecture and features, refer to the following Redbooks:

- *AIX 5L Differences Guide*, SG24-5765
- *AIX 5L Workload Manager (WLM)*, SG24-5977

8.1.7 AIX Reliability, availability, and serviceability

AIX offers customers the convenience of purchasing a complete solution from IBM (hardware, operating system, and software products). The AIX operating system is intensively tested with the RS/6000 platform. Hardware and software are extremely integrated. This results in very high reliability availability of the total solution.

AIX and the RS/6000 are tightly integrated because both products come from a single source - IBM. Integration does not need to be done by the customer. There is only one contact point for the entire solution. The following sections describe the RAS features provided by AIX and the RS/6000 platform.

8.1.7.1 Protected subsystems

In AIX, all user programs run in the user mode, which is totally separate from the kernel mode. User programs are executed in a separate address space and cannot interfere with each other or with the kernel or kernel extensions.

8.1.7.2 Dynamic Kernel

Adding a device driver to the system does not require any recompiling of the kernel as in traditional UNIX systems. Device drivers and any kernel extensions can be added on the fly without rebooting the system. Also, updates to the kernel can be applied dynamically, and kernel parameters can be changed dynamically. The kernel itself needs never to be recompiled. This is also the reason why, unlike traditional UNIX systems, a C compiler is not shipped with AIX.

8.1.7.3 Logical Volume Manager

AIX’s LVM allows for dynamic extension to the size of an active file system without any reboot and while the file system is in normal use. Doing so is totally transparent to active users of the system. LVM also offers the possibility of mirroring for extended data protection.

8.1.7.4 Journaled File System

The AIX Journaled File System, available since 1990, is a stable and recoverable file system. All transactions operated in the file system structure are logged in a Logical Volume. In case of a system crash, all non-committed transactions are rolled back so the file system is always consistent.

8.1.7.5 Journaled File System 2

The Journaled File System 2 (JFS2) is an enhanced and updated version of the JFS on AIX Version 4.3 and previous releases. JFS2 and JFS are native to the AIX 5L operating system. JFS2 provides improved structural consistency and recoverability and much faster restart times than non-journaled file systems. These file systems rely on restart time utilities (for example, fsck), which examine all of a file system's metadata (for example, directories and disk addressing structures) to detect and repair structural integrity problems. This is a time-consuming and error prone process which, in the worst case, can lose or misplace data. In contrast, JFS2 uses techniques originally developed for databases to log information about operations performed into file system metadata as atomic transactions. In the event of a system failure, a file system is restored to a consistent state by replaying the log and applying log records for the appropriate transactions. The recovery time associated with this log-based approach is much faster, since the replay utility need only examine the log records produced by recent file system activity, rather than examine all file system metadata.

8.1.7.6 AIX Backup

In AIX, the system administrator can create a bootable and installable system backup of the rootvg on tape or CD-ROM. In case of a total disk drive failure, the system can be booted from the backup media and reinstalled. No additional configuration is necessary.

The online backup possibility of logical volumes and the journaled file system increases the high availability of the system.

8.1.7.7 AIX surveillance

On systems equipped with a *Service Processor*, it is possible to enable a surveillance daemon (`survd`). This daemon runs a heartbeat with the service processor. If the AIX system hangs up for any reason, the service processor will shut down the system and reboot.

8.1.7.8 Auto Restart

In case of a system crash, the system can be automatically rebooted using the autostart=true kernel option. After the dump of system memory to disk, the system is automatically rebooted.

8.1.7.9 Remote reboot

All AIX systems can be rebooted remotely by using the `shutdown -r` command or the `reboot` command. These commands can be executed on the system through a telnet or an rlogin connection if the machine is connected to a TCP/IP network.

On RS/6000 SMP servers, the system can be booted remotely even if the power is off. This is possible because the service processor is always powered on, and remote access to the service processor is possible from a dumb terminal or via modem (if enabled).

8.1.7.10 Remote diagnostics

Most of the AIX diagnostics are online. This means that diagnostics can be executed from AIX using the `diag` command while the system is operational. However, some diagnostics dealing with network adapters require you to boot in service mode. On RS/6000 servers equipped with a service processor, hardware diagnostics can be run remotely from an ASCII terminal after a system shutdown.

8.1.7.11 Maintenance Mode

AIX allows the system administrator to boot the system in maintenance mode; however, a bootable tape or CD-ROM is required. A maintenance menu allows the administrator to access the root volume group and mount the file systems belonging to that volume group in order to access all system files. Editing files that might cause a problem is possible in this maintenance mode. This is also a good way to troubleshoot severe system problems.

8.1.7.12 Service Director/6000

IBM offers Service Director free of charge for systems that are under a maintenance contract or under warranty. It monitors the system and sends all severe errors that could occur on the system to IBM via modem connection.

8.1.7.13 AIX Error Logging Subsystem

AIX contains an extensive error logging and reporting facility. The error-logging subsystem records hardware and software failures in the error log for information purposes or for fault detection and corrective action.

The AIX operating system makes extensive use of the error-logging subsystem to warn the system administrator of serious or potentially serious hardware and software problems. Application programs can also utilize this facility by using specialized system calls within the code.

The error logging subsystem consists of three main components:

- **Error Logging** - This component is used to create the files containing error templates, to create error messages, and to insert error logging code into programs.Startinf from AIX 5L, it is allowed to control over interval in which successive duplicate errors are not recorded to log. With the following command you would therefore increase the time threshold to 1 second and the number of duplicates after the same error would again be count as a new one to 100000.

  ```
  # /usr/lib/errdeamon -m 100000 -t 1000
  ```

- **Error Processing** - This component handles the actual logging of an error when it occurs. These entries are written, in binary format, to /var/adm/ras/errlog and include a fine-grained time stamp. The error logging daemon, `errdemon`, reads error records from the /dev/error file and creates error log entries in the system error log. Besides writing an entry to the system error log each time an error is logged, the error-logging daemon performs error notification as specified in the error notification database.
- **Error Log File Processing** - This component is used by system administrators or IBM service personnel while diagnosing system problems. SMIT or the AIX command line can be used to perform error log file processing.

Error Logging Functions

The error log provides the following functions:

- It allows user or kernel functions to log errors or messages.
- It provides categories and classifications for different errors. This allows the administrator to quickly decide if an error is serious or only informational.

  The four main categories are:

  - **Hardware** - This includes device failures, such as disk drives and adapters.
  - **Software** - This includes application program failures, system program failures, and kernel problems.
  - **Operator** - This includes notification errors that get logged when the `errorlogger` command is issued.
  - **Undetermined** - This is the basic *other* category for errors that the system cannot classify into another existing category.

The six main classifications are:

  - **Permanent (PERM)** - An error from which there is no recovery
  - **Impending (PEND)** - Impending loss of availability of a device or component
  - **Temporary (TEMP)** - An error that was recovered
  - **Informational (INFO**) - An informational error log entry that may have been put there from a user's program
  - **Performance (PERF)** - Indicates that the performance of a device or a component has degraded below an acceptable level
  - **Unknown (UNKN)** - Severity could not be determined

- It allows a custom template for each error log type. The format of each error log entry is different depending on the type of error.
- It provides automatic notification for selected errors. A notification method allows an executable to be run when certain errors are detected. For example, this might be to send an e-mail to the administrator when a user application core dumps.
- It provides a formatter to browse the error log. The `errpt` command allows the user to format and browse the log entries.
- It provides analysis of error logs by diagnostics. Diagnostics can read and analyze the error log and perform diagnostics on offending items.

Link between error log and diagnostics

Starting with AIX 5L the information generated by the diag program is pub back into the error log entry, so that it is easy to make the connection between the error event and, for instance, the FRU number needed to replace failing hardware.

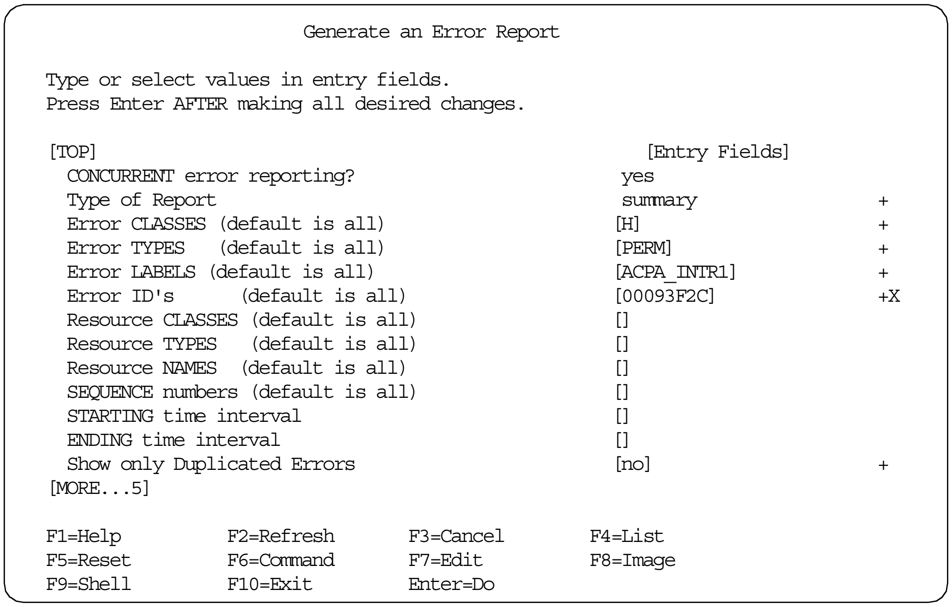
Reading the error log

In AIX, there are several methods available to view the contents of the error log. Any user on the system may view the error log content. Only the system administrator can remove entries from the error log.

The error log is circular, meaning it has a finite length before overwriting the oldest messages. The administrator can change the total length of the log or clear it out from time to time.

The error log can be interrogated from the AIX command line with the `errpt` command. The `errpt` command has many flags or switches to allow the user to filter data or expand the level of detail shown. Most commonly, it is used in the form, `errpt -a`, which shows the complete error log, all entries in expended format. Starting from AIX 5L, `errpt` command now also supports and intermediate format with the -A flag, in addition to summary and detail.

The SMIT interface can also be used to query the error log. The `smit errpt` fastpath can be used to access the error log menu directly. For example, to show all information messages, the system administrator would enter parameters as shown in below screen.



```
                        Generate an Error Report

 Type or select values in entry fields.
 Press Enter AFTER making all desired changes.

 [TOP]                                               [Entry Fields]
   CONCURRENT error reporting?                      yes
   Type of Report                                   summary            +
   Error CLASSES (default is all)                  [H]                 +
   Error TYPES   (default is all)                  [PERM]              +
   Error LABELS (default is all)                   [ACPA_INTR1]        +
   Error ID's      (default is all)                [00093F2C]          +X
   Resource CLASSES (default is all)               []
   Resource TYPES   (default is all)               []
   Resource NAMES  (default is all)                []
   SEQUENCE numbers (default is all)               []
   STARTING time interval                          []
   ENDING time interval                            []
   Show only Duplicated Errors                     [no]                +
 [MORE...5]

 F1=Help          F2=Refresh       F3=Cancel        F4=List
 F5=Reset         F6=Command       F7=Edit          F8=Image
 F9=Shell         F10=Exit         Enter=Do
```

If there were such errors on the system, the administrator would receive results similar to those shown in the following screen:

```
---------------------------------------------------------------------------
LABEL:          STOK_RCVRY_EXIT
IDENTIFIER:     5BF9FD4D

Date/Time:       Fri Feb 16 18:29:28 CST
Sequence Number: 193
Machine Id:      000FA16D4C00
Node Id:         rs9916a
Class:           H
Type:            TEMP
Resource Name:   tok0
Resource Class:  adapter
Resource Type:   14101800
Location:        1P-08
VPD:
        Loadable Microcode Level.......WW18CB
        Part Number.................00000000
        EC Level....................00D51237
        Serial Number...............00432064
        FRU Number..................00000000
        Manufacturer................IBM982
        Network Address.............0004AC6173FE
```

8.1.7.14 The syslog daemon

Another error-logging facility that exists in AIX is called `syslog`. The `syslog` facility comes from the BSD style of UNIX. One highly-valued feature is the ability to easily reroute messages to another system via mail or files. The `syslogd` daemon reads a datagram socket and sends each message line to a destination described by the `/etc/syslog.conf` configuration file. The `syslogd` daemon reads the configuration file when it is activated and when it receives a hang-up signal. Messages can be rerouted to the error logging facility and vice versa.

Like the error-logging subsystem, the syslog facility classifies events in the form Facility, Level, Destination.

A *Facility* is a subsystem that sends a message to be logged. Although there are many *Facilities*, the major ones are:

- **kern** The kernel
- **mail** The mail subsystem
- **lpr** The printing subsystem
- **auth** The login authentication systemg facility
- **user** User level
- * All facilities

The *Level* is the relative severity of the message. Typical severity levels are, in order of decreasing seriousness:

- **emerg** Emergency. Usually a system panic or other fatal disaster
- **alert** A serious error needing immediate attention
- **crit** Critical errors; typically hardware errors
- **err** General errors from user programs
- **warning** A warning, not likely to be fatal
- **info** Informational messages

In addition, there is a special level called debug that can be used to help debug an application or subsystem. Debug should only be used for temporary problem determination because it tends to generate a lot of data in a short amount of time. The Destination is used to determine where to route or log this error. This could be a log file on the local machine, a remote machine, another program, or an e-mail message.

8.1.7.15 Resource Monitoring and Control (RMC)

In AIX 5L, a new Resource Monitoring and Control (RMC) subsystem is available that is comparable in function to the Reliable Scalable Cluster Technology (RSCT) on the IBM SP type of machines.This subsystem allows you to associate predefined responses with predefined conditions for monitoring system resources. An example is to broadcast a message when the /tmp file system becomes 90 percent full to summon the attention of a dutiful system administrator. At the time of writing, this feature is only available on the POWER platform.

Packaging and Installation

The RMC subsystem is installed by default and is delivered in one bundle named rsct.ore containing nine different file sets as below screen.

```
# lslpp -l | grep rsct
  rsct.core.auditrm          2.1.0.0  COMMITTED  RSCT Audit Log Resource
  rsct.core.errm             2.1.0.0  COMMITTED  RSCT Event Response Resource
  rsct.core.fsrm             2.1.0.0  COMMITTED  RSCT File System Resource
  rsct.core.gui              2.1.0.0  COMMITTED  RSCT Graphical User Interface
  rsct.core.hostrm           2.1.0.0  COMMITTED  RSCT Host Resource Manager
  rsct.core.rmc              2.1.0.0  COMMITTED  RSCT Resource Monitoring and
  rsct.core.sec              2.1.0.0  COMMITTED  RSCT Security
  rsct.core.sr               2.1.0.0  COMMITTED  RSCT Registry
  rsct.core.utils            2.1.0.0  COMMITTED  RSCT Utilities
  rsct.core.rmc              2.1.0.0  COMMITTED  RSCT Resource Monitoring and
  rsct.core.utils            2.1.0.0  COMMITTED  RSCT Utilities
```

All executables and related items are installed into the /usr/sbin/rsct directory, while the log files and other temporary data is located in /var/ct. The following entry is located in /etc/inittab:

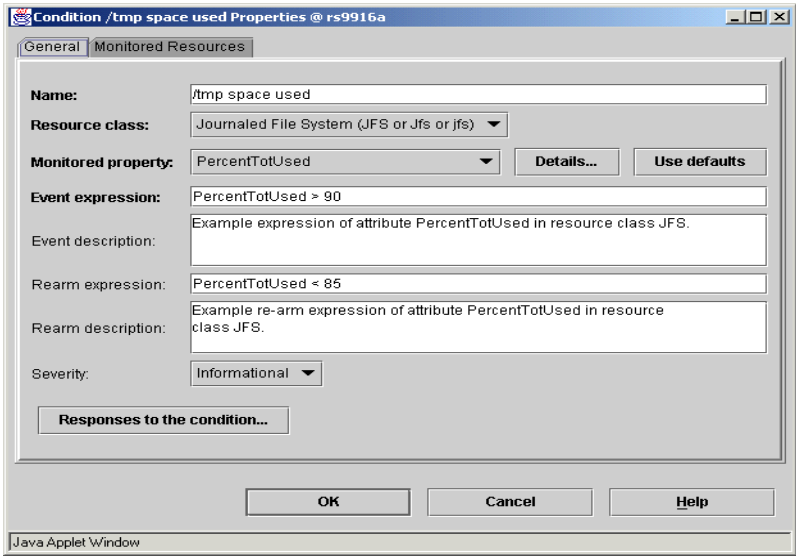
```
ctrmc:2:once:/usr/bin/startsrc -s ctrmc > /dev/console 2>&1
```

Due to this entry, the RMC subsystem is also automatically started. This subsystem can be controlled using the SRC commands, but it also has its own control command (/usr/sbin/rsct/bin/rmcctrl), which is the preferred way to stop and start it. Due to the number of available options on this subsystem, it can only be controlled through the Web-based System Manager. A SMIT interface is not available at the time of publication.

Concepts of RMC

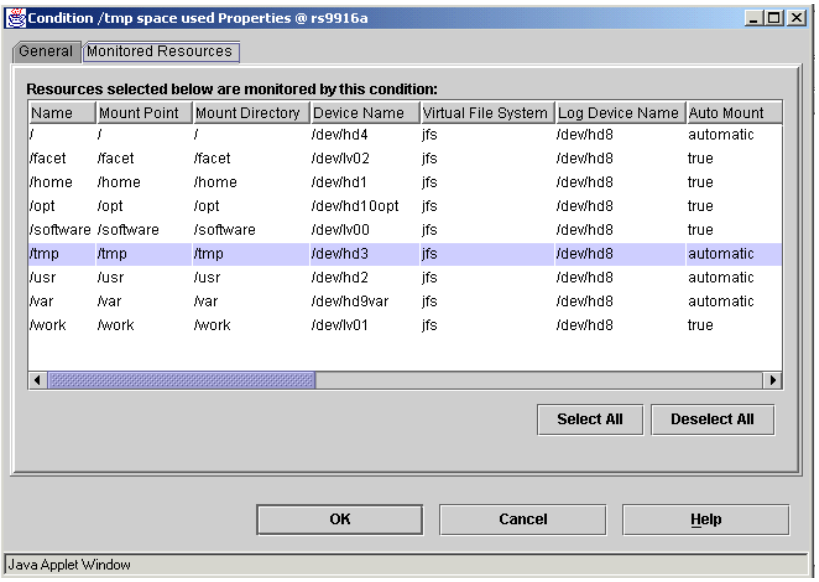
The basic function of RMC is based on two concepts: conditions and responses. To provide you a ready-to-use system, 84 conditions and eight responses are predefined for you. You can use them as they are, customize them, or use them as templates to define your own conditions and responses.

To monitor a condition, simply associate one or more responses with the condition. A condition monitors a specific property, such as total percentage used, in a specific resource class, such as JFS. You can monitor the condition for one, or more, or all the resources within the monitored property, such as /tmp, or /tmp and /var, or all the file systems. Each condition contains an event expression to define an event and an optional rearm expression to define a rearm event. The event expression is a combination of the monitored property, mathematical operators, and some numbers, such as PercentTotUsed > 90 in the case of a file system. The rearm expression is a similar entity; for example, PercentTotUsed < 85. Figure 86 on page 264 and Figure 87 on page 264 provide an example for a condition definition panel and monitored resource definition panel using Web-based System Manager, respectively.



Name:	/tmp space used
Resource class:	Journaled File System (JFS or Jfs or jfs)
Monitored property:	PercentTotUsed
Event expression:	PercentTotUsed > 90
Event description:	Example expression of attribute PercentTotUsed in resource class JFS.
Rearm expression:	PercentTotUsed < 85
Rearm description:	Example re-arm expression of attribute PercentTotUsed in resource class JFS.
Severity:	Informational

Figure 86. RMC General Resource tab



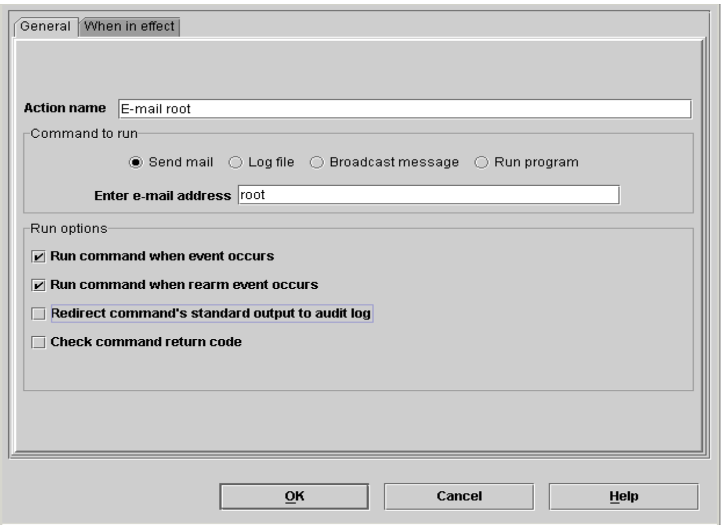
Resources selected below are monitored by this condition:

Name	Mount Point	Mount Directory	Device Name	Virtual File System	Log Device Name	Auto Mount
/	/	/	/dev/hd4	jfs	/dev/hd8	automatic
/facet	/facet	/facet	/dev/lv02	jfs	/dev/hd8	true
/home	/home	/home	/dev/hd1	jfs	/dev/hd8	true
/opt	/opt	/opt	/dev/hd10opt	jfs	/dev/hd8	true
/software	/software	/software	/dev/lv00	jfs	/dev/hd8	true
/tmp	/tmp	/tmp	/dev/hd3	jfs	/dev/hd8	automatic
/usr	/usr	/usr	/dev/hd2	jfs	/dev/hd8	automatic
/var	/var	/var	/dev/hd9var	jfs	/dev/hd8	automatic
/work	/work	/work	/dev/lv01	jfs	/dev/hd8	true

Figure 87. RMC Monitored Resources tab

If the event expression becomes true, the Event Response Resource Manager (ERRM) checks all responses connected to this monitor and executes the corresponding event actions.

Figure 88 shows an example of a response definition panel; you can define the response action when the condition is met.



Action name E-mail root

Command to run

● Send mail ○ Log file ○ Broadcast message ○ Run program

Enter e-mail address root

Run options

☑ **Run command when event occurs**

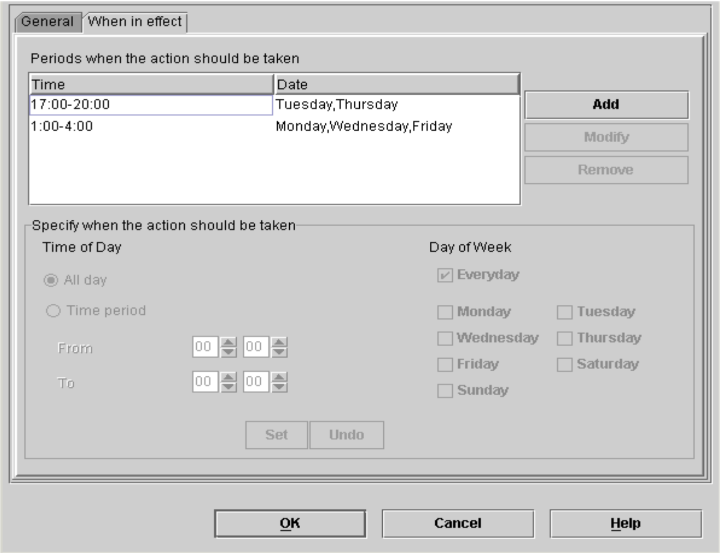
☑ **Run command when rearm event occurs**

□ **Redirect command's standard output to audit log**

□ **Check command return code**

Figure 88. RMC - General Response Panel

You can also specify a time window in which the action is active, such as always, or only during on-shift on weekdays as Figure 89.



Periods when the action should be taken

Time	Date
17:00-20:00	Tuesday,Thursday
1:00-4:00	Monday,Wednesday,Friday

Specify when the action should be taken

Time of Day

● All day

○ Time period

From 00 00

To 00 00

Day of Week

☑ Everyday

□ Monday □ Tuesday

□ Wednesday □ Thursday

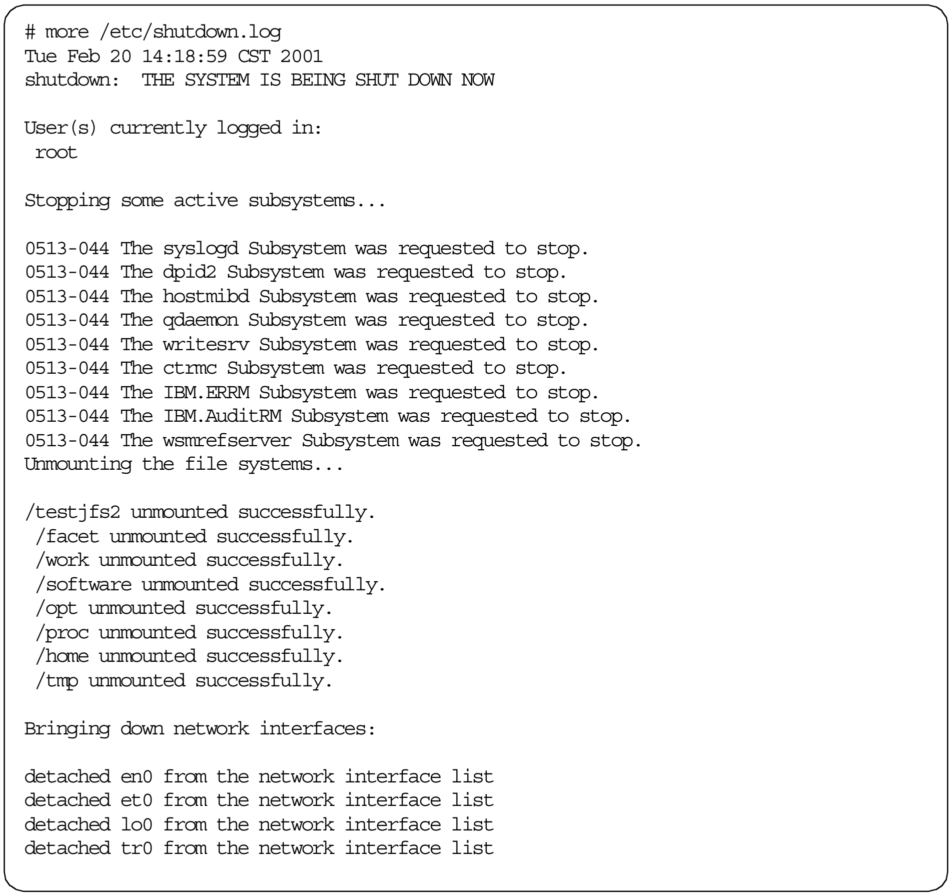
□ Friday □ Saturday

□ Sunday

Figure 89. RMC - When in effect panel

8.1.7.16 Shutdown logging

AIX 5L enhanced the `shutdown` command with a -l flag to log the output (from select actions during the shutdown) to the file /etc/shutdown.log. This log is available even if there are problems with booting the system and the machine had to be shutdown several time. The contents of this file appears similar to the following screen.



```
# more /etc/shutdown.log
Tue Feb 20 14:18:59 CST 2001
shutdown:  THE SYSTEM IS BEING SHUT DOWN NOW

User(s) currently logged in:
 root

Stopping some active subsystems...

0513-044 The syslogd Subsystem was requested to stop.
0513-044 The dpid2 Subsystem was requested to stop.
0513-044 The hostmibd Subsystem was requested to stop.
0513-044 The qdaemon Subsystem was requested to stop.
0513-044 The writesrv Subsystem was requested to stop.
0513-044 The ctrmc Subsystem was requested to stop.
0513-044 The IBM.ERRM Subsystem was requested to stop.
0513-044 The IBM.AuditRM Subsystem was requested to stop.
0513-044 The wsmrefserver Subsystem was requested to stop.
Unmounting the file systems...

/testjfs2 unmounted successfully.
 /facet unmounted successfully.
 /work unmounted successfully.
 /software unmounted successfully.
 /opt unmounted successfully.
 /proc unmounted successfully.
 /home unmounted successfully.
 /tmp unmounted successfully.

Bringing down network interfaces:

detached en0 from the network interface list
detached et0 from the network interface list
detached lo0 from the network interface list
detached tr0 from the network interface list
```

8.1.7.17 System dump facility

When a severe error occurs and the system cannot guarantee the consistency of its state, it stops all activity on the system and initiates a system dump. System dumps can also be initiated by users with root authority. A system dump creates a picture of your system’s memory contents. System administrators and programmers can generate a dump and analyze its contents when debugging new applications.

After the system dump occurs, there is a flashing 888 on the display, or the system reboots automatically if it is so configured. By default, the dump file is created on a logical volume dedicated for paging space, but the system administrator can create a separate logical volume and configure it as a dump device. If the dump is created on /dev/hd6 (paging space) while the system is booting, the dump is copied to the /var/adm/ras/vmcore.x file. If there is not enough space in the /var file system (note that the dump can be quiet large), the system operator is prompted on console for inserting tape into the tape drive and copying the dump to tape. It is very important that the system administrator do so at that time because the dump cannot be collected later, once the paging space is initialized and used by the system for paging. After collecting the dump, the system administrator should collect all information about his/hers system with the `snap -a` command and send the tape to the IBM support center to determine the reason for the dump.

The system dump can also be initiated via the command line (with the `sysdumpstart` command) or via SMIT if the system administrator or support center wants to analyze it. On older RS/6000 systems, it is possible to turn the key to the *Service* position and press the yellow Reset button once to initiate a dump. If the system is not responding normally (the administrator cannot log on), but interrupts are still enabled (keyboard interrupt), it is possible to initiate a dump by pressing Ctrl+Alt+NumPad 1 on the graphical console.

The dump can by analyzed by the *crash* interactive tool.

Starting from AIX 5L, following enhancements are provided in the area of system dumps:

- A new command, dumpcheck, checks if the dump device and the copy directory for the dump are large enough to actually accept a system dump. Below screen shows an example of `dumpchek` command.

```
# dumpcheck -p
There is not enough free space in the file system containing the copy directory
to accommodate the dump.
File system name         /var/adm/ras
Current free space in kb         1068
Current estimated dump size in kb        97280
```

- The creation of a core file for a process without terminating the process. An application can now create a core file by using the new coredump() system call. This call takes, as a single parameter, a pointer to a coredumpinfop structure that sets the path and file name for the core file to be generated.

8.1.7.18 System Hang Detection

AIX 5L provides a SMIT-configurable mechanism to detect system hangs and initiate the configured action. It relies on a new daemon named shdaemon and a corresponding configuration program named `shconf`.

In the case where applications adjust their process or thread priorities using system calls, there is the potential problem that their priorities will become so high that regular system shells are not scheduled. In this situation, it is difficult to distinguish a system that really hangs from a system that is so busy that none of the lower priority tasks, such as user shells, have a chance to run. The new system hang detection feature uses a shdaemon entry in the /etc/inittab file with an action field that is set to off by default. Using the `shconf` command or SMIT (fastpath shd), you can enable this daemon and configure the actions it takes when certain conditions are met. The following flags are allowed with the `shconf` command.

```
shconf [ -d ][ -R |-D [ -O] | -E [ -O ] | [[ -a Attribute ] ..] -l prio [-H]
```

The -d flag displays the current status of the shdaemon. The -R flag restores the system default values. With the -D and -E flags, you can display either the default or the effective values of the configuration parameters. The -H flag adds an optional header to this output. The screen below shows an example of `shconf` command.

```
# shconf -d
sh_pp=disable
# shconf -E -l prio -H
attribute  value        description

sh_pp      disable      Enable Process Priority Problem
pp_errlog  disable      Log Error in the Error Logging
pp_eto     2            Detection Time-out
pp_eprio   60           Process Priority
pp_warning disable      Display a warning message on a console
pp_wto     2            Detection Time-out
pp_wprio   60           Process Priority
pp_wterm   /dev/console Terminal Device
pp_login   enable       Launch a recovering login on a console
pp_lto     2            Detection Time-out
pp_lprio   56           Process Priority
pp_lterm   /dev/tty0    Terminal Device
pp_cmd     disable      Launch a command
pp_cto     2            Detection Time-out
pp_cprio   60           Process Priority
pp_cpath   /            Script
pp_reboot  disable      Automatically REBOOT system
pp_rto     5            Detection Time-out
pp_rprio   39           Process Priority
```

8.1.7.19 Tape Backup/restore support

AIX includes a system backup facility (accessible via SMIT) that allows the system administrator to create a bootable image of the root volume group. In case of a system crash and if the system cannot be recovered in other ways, it is always possible to boot the system on the backup tape and restore the entire root volume group. Non-root volume groups will be preserved and accessible after the restore process is completed.

The AIX backup utility allows different kinds of backups. Refer to section 8.1.5.1, “AIX backup tools” on page 248, for more information on AIX backup processes.

8.1.7.20 UPS support

The Uninterruptable Power Supply (UPS) is supported on AIX. On a rack system, the UPS is a feature that can be purchased directly from IBM.

UPS can be connected to one of the serial ports of an RS/6000 system and the special software that comes with most UPSs can be configured to initiate a `shutdown` command after the main power is lost in order to stop the applications and halt the system in a regular way. If the UPS is large enough, the RS/6000 can continue operation until the main power comes back.

8.1.7.21 RAID support

As explained in section 6.1.6, “AIX 5L and RAID support” on page 131, AIX supports mirroring and striping. In addition, IBM provides a large range of disk units and controllers to enable RAID 5.

8.1.8 AIX printing

Regardless of the concept of paperless offices, the printing is still a very important issue in modern IT environments. Both operating systems covered here directly support large number of different printers, from matrix and ink-jet to laser ones.

From AIX, we can print to printers attached to a local RS/6000 system, to printers attached to other UNIX or other operating systems, or to Network Printers (Printers attached directly to the LAN).

For more details about AIX 5L printing, refer to the IBM Redbook, *Printing for Fun and Profit under AIX 5L*, SG24-6018.

8.1.8.1 AIX printing terminology

When configuring a printing subsystem on AIX, the system administrator must be familiar with the following terms:

Printer
This is an actual physical device for printing.

Queue
Users submit their jobs (files) to queues. The print spooler subsystem then sends the data to the printer. For more details, see section 8.1.8.2, “AIX printing process” on page 271.

Virtual printer
A virtual printer is a single logical view of the components that make up a printer from a user’s perspective. A virtual printer describes the combination of a printer device, a data stream that a printer supports, and the specific configuration of the queue and queue device that serves the virtual printer. The virtual printer is associated with a print queue. It is possible to define a print queue for each data stream that the printer supports so that several queues point to one physical printer. It is also possible to configure one print queue to points to many physical printers (see Figure 90 on page 271).

Spooling
This is the concept of submitting the jobs for processing and handling them in a controlled predefined way. They are two basic methods: FCFS (First Come First Serve) or SJN (Shortest Job Next).

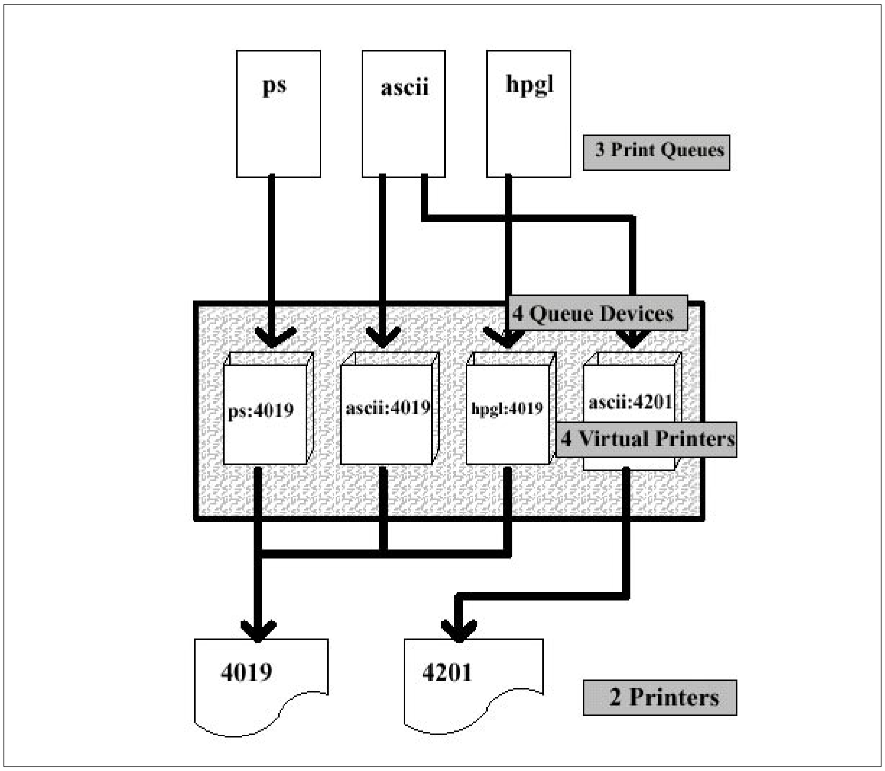


Figure 90. Concept of virtual printers

8.1.8.2 AIX printing process

Figure 91 on page 272 depicts the AIX printing process.

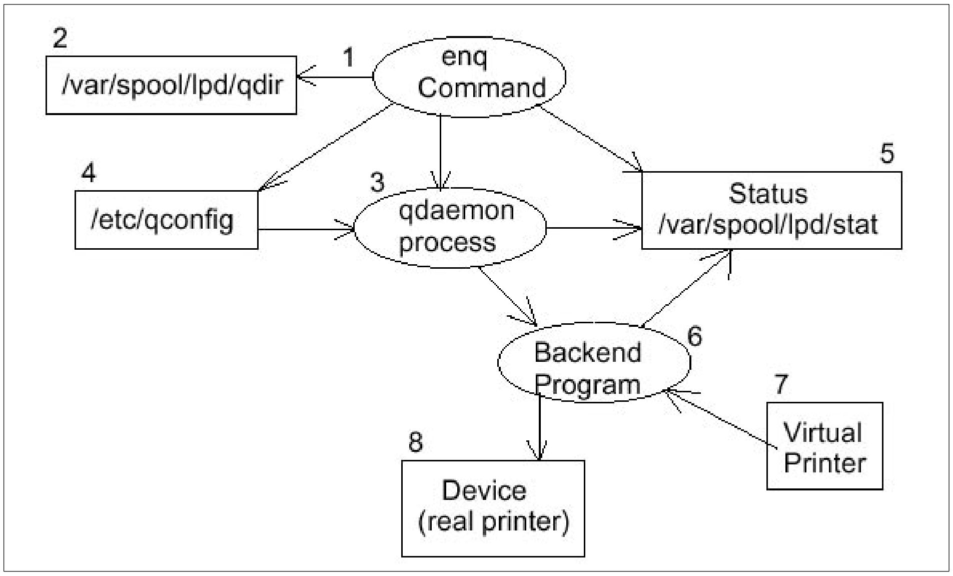


Figure 91. AIX printing process

The following steps are performed by the AIX operating system:

1. The user submits a job with one of many AIX commands, such as `lpr`, `qprt`, `enq`, or `lp`. All of them are actually only front-ends for the `enq` command.
2. The particular command translates its options into `enq`'s options and calls the `enq` command.
3. The `enq` command checks to see if the queue name desired is a valid queue in the /etc/qconfig file. If so, it continues; if it is not valid, an error message is given to the user.
4. The `enq` command puts an entry in the /var/spool/lpd/qdir directory identifying the job to be run. Note that the original file is used. To copy the file to the spool directory for printing, you need to supply an option to the command you use to submit the print job.
5. The `qdaemon` is notified of a new job in its /var/spool/lpd/qdir directory.
6. When the queue is ready for the job, the qdaemon picks up the information from the /etc/qconfig file describing the queue.
7. The qdaemon updates the stat file for the appropriate queue to show that the queue is now working on a new job.

8. The `qdaemon` starts the backend program (usually piobe), passing the file names and appropriate options on the command line. Information on the job status is stored and updated in the `/var/spool/lpd/stat` file for use by the `qdaemon` and backend program.
9. The back end figures out the appropriate virtual printer and merges it with the actual file.
10. The back end program sends its data stream to the device driver for the appropriate printer.

It is also possible to bypass the spooling subsystem completely and send the file directly to the special device file, /dev/lp0, and, therefore, to the printer. This option is, however, not recommended in multiuser environments because, if several users would do this at the same time, the result could be unpredictable. Also, we would loose all the formatting and configuration functionality of Virtual printers. But, sending the file directly to the printer can be very useful in determining problems on a printing subsystem; this step can help determine if the problem is with the physical printer and cable connection or with the spooling subsystem.

8.1.8.3 AIX printing management

This section describes the necessary steps for using a printing subsystem on AIX.

This step is usually done via SMIT, VSM or Web-based System Manager.

Defining the printers

In the first step, we choose if the printer is attached to our local RS/6000 system, to a remote UNIX system, or is a Network Printer.

The second step is to choose the type of the printer. Support for several IBM and HP printers is part of the Server Bundle package and is installed by default. If we would like to define other supported printers we must install additional file sets from the installation CD or tape. If we would like to use unsupported printers, we have two options:

- We can use one of the emulation modes of the printer (many printers emulate an IBM or HP printer).
- We can choose a similar kind of printer from supported printer types and change its definitions in the Virtual Printer definition file.

The third step is to choose which port the printer is connected (parallel or serial), if we have a local attached printer or IP hostname or address, and the name of the queue on the remote system if we have a printer attached to the remote host or Network Printer.

Defining the queues

In this step, we define the names of the queues. The number of queues depends of our specific needs, but, in general, each printer will have as many queues pointing to it, as different data streams it supports, such as ASCII, PostScript, PCL, HPGL, IPDS, and so on. For each data stream, we must define its own queue (if we want to use it).

After this, we can start using the printer(s). The management of an existing Printing subsystem consists of:

Queue management

This can be done by any AIX user interfaces (command line, SMIT, VSM, Web-based System Manager). We can stop and start the queues and add or delete the queues.

Print job management

This can also be done by any of the AIX user interfaces. The system administrator can see the status of print jobs, hold printing job on queue, cancel a printer job or transfer the job to another printing queue. He or she can also increase or decrease the priority of printing jobs.

Print server

If we want other systems (users) to be able to print to a printer attached locally to our RS/6000, we must start Print Server Subsystem (via command line or SMIT) and then explicitly allow each client to print to our printer.

8.1.8.4 System V Release 4 Print Subsystem

Existing AX Print Subsystem was designed to combine the features of the System V and Berkeley Software Distribution (BSD) printing standard, along with some unique features found only in AIX.

With the onset of the development of AIX5L for Itanium-based platforms, it was necessary to look for an alternative print solution that provided a standard, less complex print subsystem that potentially embodies the concept of directory enablement, and lets the source code of AIX 5L for POWER and AIX 5L for Itanium-based systems intersect as much as possible.

The AIX 5L for Itanium-based systems' development team had chosen the System V Release 4 (SVR4) print subsystem as the printing solution, and this print subsystem was added to AIX 5L for POWER. In AIX 5L for Itanium-based systems, it will be the only print subsystem offered.

If the code for both print subsystems is installed, the base operating system of the current AIX 5L release uses the traditional AIX print subsystem by default and the System V print subsystem is not active.

AIX 5L provides a command menu, a SMIT menu, and a Web-based System Manager menu, which allows the system administrator to switch between the AIX and the System V print subsystems, but will not allow both print subsystems to be active at the same time.

Features

System V Release 4 (SVR4) Printing subsystem supports following features.

- Local printing (parallel and serial)
- Remote printing using BSD'd lpd protocol (RFC 1179)
- Network printing using HP's JetDirect

Installation package

The installation package of the System V print subsystem is consisted of following file sets and included in OS CD.

- bos.rte.printers
- printers.rte
- printers.msgs.xx_XX.rte (locale specific)
- bos.svprint

SVR4 print subsystem process logic

Each print request is sent to a spooling daemon (lpsched) that keeps track of all the jobs. The daemon is created when you start the print service. The spooling daemon is also responsible for keeping track of the status of the printers and slow filters: when a printer finishes printing a job, the daemon starts printing another job if one is queued.

You can customize the print service by adjusting or replacing some of the items noted (the following numbers are explanations of the keys used in the Figure 92 on page 276).

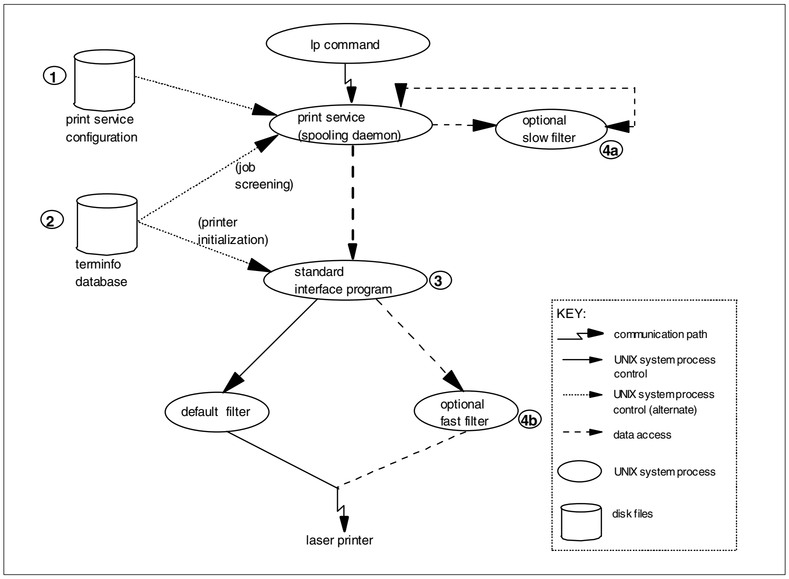


Figure 92. SVR4 print subsystem process logic

1. For most printers, you need only to change the printer configuration stored on disk. For further details refer to the lpadmin command documentation for adding or modifying a local printer.
2. The print service relies on the standard interface script and the terminfo database to initialize each printer and set up a selected page size, character pitch, line pitch, and character set. For printers that are not represented in the terminfo database, you can add a new entry that describes the capabilities of the printer. The print service uses the terminfo database in two parallel capacities: screening print requests to ensure that those accepted can be handled by the desired printer, and setting the printer so it is ready to print the requests. For instance, if the terminfo database does not show a printer capable of setting a page length requested by a user, the spooling daemon rejects the request.However, if it does show it to be capable, then the interface program uses the same information to initialize the printer.
3. If you have a particularly complicated printer or if you want to use features not provided by the print service, you can change the interface script. This script is responsible for managing the printer: it prints the banner page, initializes the printer, and invokes a filter to send copies of the user's files to the printer.

4. To provide a link between the applications used on your system and the printers, you can add slow and fast filters. Each type of filter can convert a file into another form, for example, mapping one set of escape sequences into another, and can provide a special setup by interpreting print modes requested by a user. Slow filters are run separately by the spooling daemon to avoid tying up a printer. Fast filters are run so their output goes directly to the printer; thus, they can exert control over the printer.

SVR4 printing subsystem user commands

Figure 18 shows the commands that are available to all users.

Table 18. SVR4 printing subsystem user commands

Command	Description
cancel	The cancel command allows users to cancel print requests previously sent with the lp command. The command permits cancellation of requests based on their request-ID or based on the login-ID of their owner.
lp	The lp command arranges for the named files and associated information to be printed. Alternatively the lp command is used to change the options for a request submitted previously.
lpstat	The lpstat command displays information about the current status of the print service. If no options are given, lpstat displays the status of all print requests made by the user.

SVR4 printing subsystem administrative commands

Figure 19 shows the commands that are available only to an administrator.

Table 19. SVR4 printing subsystem administrative commands

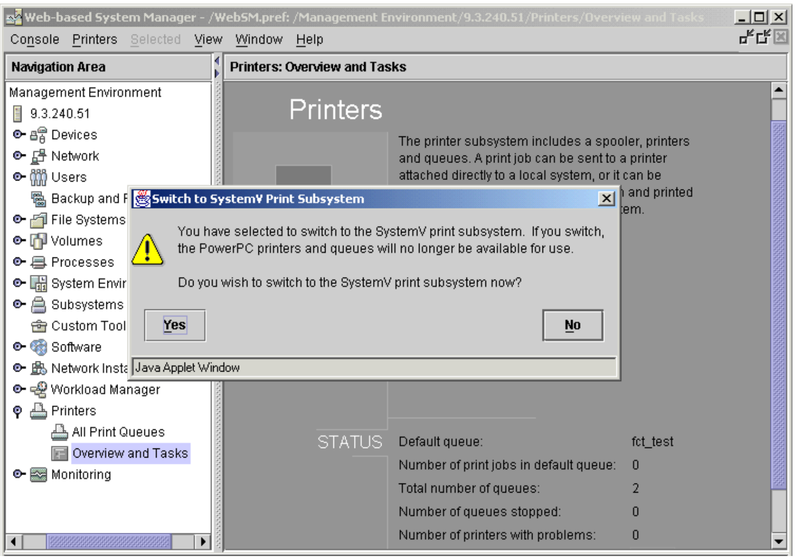
Command	Description
accept/reject	accept allows the queuing of print requests for the named destinations. A destination can be either a printer or a class of printers. reject prevents queuing of print requests for the named destinations.
enable/disable	The enable command activates the named printers, enabling them to print requests submitted by the lp command. If the printer is remote, the command will only enable the transfer of requests to the remote system.
lpadmin	lpadmin configures the LP print service by defining printers and devices. It is used to add and change printers, to remove printers from the service, to set or change the system default destination, to define alerts for printer faults, to mount print wheels, and to define printers for remote printing services.

Command	Description
lpfilter	The lpfilter command is used to add, change, delete, and list a filter used with the LP print service. These filters are used to convert the content type of a file to a content type acceptable to a printer.
lpforms	The lpforms command is used to administer the use of preprinted forms, such as company letterhead paper, with the System V print service.
lpmove	lpmove moves requests that were queued by lp between destinations (printers or classes of printers).
lpsched	lpsched allows you to start the System V print service.
lpshut	lpshut shuts down the print service.
lpsystem	The lpsystem command is used to define parameters for the LP print service, with respect to communication with remote systems.
lpusers	The lpusers command is used to set limits to the queue priority level that can be assigned to jobs submitted by users of the System V print service.

Switching between printing systems

To switch between printing subsystem, you can use one of following two methods like:

- On Web-based System Manager, you select **Printers** -> **Overview and Tasks** -> **Switch to SystemV print subsystem**; you should see a window as shown in Figure 93 on page 279.



The printer subsystem includes a spooler, printers and queues. A print job can be sent to a printer attached directly to a local system, or it can be … and printed … tem.

You have selected to switch to the SystemV print subsystem. If you switch, the PowerPC printers and queues will no longer be available for use.

Do you wish to switch to the SystemV print subsystem now?

Default queue: fct_test
Number of print jobs in default queue: 0
Total number of queues: 2
Number of queues stopped: 0
Number of printers with problems: 0

Figure 93. Switching to System V print subsystem

- You can run switch.prt script, which supports two flags (-s: to select either AIX or System V print subsystems, -d to display the current active system).

8.1.9 AIX terminal support

Terminal (ASCII terminal) access has been supported on AIX right from the beginning of the AIX operating system. Actually, even today, the largest RS/6000 servers are often not installed with graphical adapters and graphical terminals and use a plain ASCII terminal as the system console.

8.1.9.1 Physical connection of terminals

Each RS/6000 has at least two serial ports (some have three) on which we can connect ASCII terminals. One of the ports is usually used as a system console.

If we would like to use more ASCII terminals, we can install an 8-port or 128-port Async adapter/controller in RS/6000 system. Note that a 128-port controller uses an additional 16-port RANs (Remote Async Node); so, we can configure it as a 16, 32, 48, (and so on) subsystem.

8.1.9.2 Configuration of terminals

After we physically install the terminals, we must define them to AIX. It is strongly recommended that you use the SMIT or Web-based System

Manager tool for this task because of the large number of parameters that can be used for this task.

The main parameters to define are:

- Physical connection address (port)
- Baud rate (default is 9600)
- Login characteristics (login enabled/disabled)
- Type of the terminal (emulation, ibm3153, vt100, vt220 and so on)
- Serial line characteristics (parity, bits per character, stop bits, and so on)

After this step, the users, if they have valid user account, will be able to log on the system.

Note

The system administrator can configure the user account in such a way that the AIX shell (ksh) will be started each time the user logs on (this is the default), or the system administrator can define some other application to be started immediately after logon. In this way, the user will never actually see the AIX user interface on his or her terminal.

This is sometimes done for security reasons and sometimes for convenience reasons, that is, ease of use.

8.1.10 AIX support for national languages

In this section, we will cover the support for languages other than English on the AIX operating system.

Support for national languages consists of:

- Translation of operating system user interface, menus, system applications, system tools, and help files.
- Translation of hardcopy and softcopy documentation.
- Support for code-pages.
- Support for keyboards.
- Support for formatting the way the time, date, and currency symbols are represented in a particular country.

The AIX operating system is translated (system messages, user interfaces, help files) into the following languages:

- Brazilian Portuguese
- Catalan
- Czech
- French (Universal)
- German (Universal)
- Hungarian
- Italian
- Japanese-Kanji
- Korean
- Polish
- Russian
- Spanish
- Swedish
- Simplified Chinese
- Traditional Chinese
- U.S. English

8.1.10.1 AIX locales

Support for locales is the way characters and time, date, and currency symbols are represented by the operating system and is provided additionally for the following countries/regions:

- Albanian
- Arabic
- Belgian Dutch
- Belgian French
- Brazilian Portuguese
- Bulgarian
- Canadian French
- Catalan
- Croatian
- Czech
- Danish
- Dutch (Netherlands)
- English - Australia
- English - Belgium
- English - South Africa
- English U.K.
- English U.S.
- Estonian
- Finish
- French
- German

- Greek
- Hebrew
- Hindi
- Hungarian
- Icelandic
- Italian
- Italian - Switzerland
- Japanese (Kanji)
- Korean
- Latvian
- Lithuanian
- Macedonian
- Norwegian
- Polish
- Portuguese
- Romanian
- Russian
- Serbian Cyrillic
- Serbian Latin
- Simplified Chinese
- Slovak
- Slovene
- Spanish
- Swedish
- Swiss French
- Swiss German
- Traditional Chinese
- Turkish

8.1.10.2 AIX Supported Code Pages

AIX 5L is capable of concurrently handling on a per-process basis any one of the following:

- ISO 8859-1 (Latin 1 - Western European)
- ISO 8859-2 (Latin 2 - Eastern European)
- ISO 8859-3 (Latin 3)
- ISO 8859-4 (Latin 4)
- ISO 8859-5 (Latin/Cyrillic)
- ISO 8859-6 (Latin/Arabic)
- ISO 8859-7 (Latin/Greek)
- ISO 8859-8 (Latin/Hebrew)
- ISO 8859-9 (Turkish)
- Extended UNIX Code for Japanese

- Extended UNIX Code for Korean
- Extended UNIX Code for Traditional Chinese
- IBM PC Code Set 850 (Latin 1)
- IBM PC Code Set 932 (Japanese)
- IBM PC Code Set 1046 (Arabic)
- IBM PC Code Set 856 (Hebrew)

AIX supports the Euro symbol for following *Locales*:

- Catalan
- Dutch (Belgium)
- Dutch
- Finnish
- French (Belgium)
- French
- German
- Italian
- Portuguese
- Spanish

8.2 Windows 2000 System Management

In this chapter, we will cover different user interfaces available for system administrators to perform common system administration tasks, such as storage management, user management, device management, network management and so on for Windows 2000.

8.2.1 Windows 2000 installation methods

The Windows 2000 installation is designed in such a way that a simple installation can be accomplished by a user with limited experience, yet it still offers some advanced features to allow customizing for more advanced users.

Windows 2000 allows a system administrator to install a new system, upgrade a system running a previous version of Windows to a Windows 2000, or install Windows 2000 on an existing Windows NT system on a separate partition and then have the ability to boot either Windows NT or Windows 2000. In the latter case, the system administrator should consider the compatibility issues of NTFS Version 4 used in Windows NT and NTFS Version 5 used in Windows 2000. For more details about Windows 2000 NTFS, see section 6.2.2, “Windows 2000 file systems” on page 137.

An upgrade to Windows 2000 Server can be done from the following Windows version:

- Windows NT version 3.51 Server
- Windows NT version 4.0 Server
- Windows NT version 4.0 Terminal Server
- Windows 2000 Server Beta 3 or final-release candidates

If you are using Windows NT Version 4.0 Server Enterprise Edition, you can upgrade to Windows 2000 Advanced Server but not to Windows 2000 Server.

An upgrade to Windows 2000 Professional can be done from the following previous desktop versions of Windows:

- Windows 95
- Windows 98
- Windows NT Workstation 3.51
- Windows NT Workstation 4.0

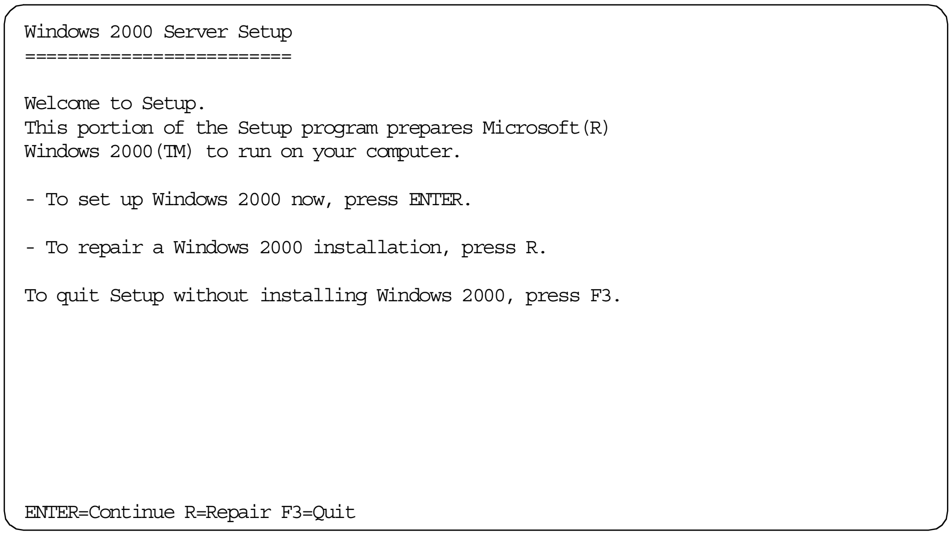
A new installation or an upgrade can be performed from CD-ROM or from the network. There is no support for installation from tape.

Windows 2000 installation consists of a text portion, where files are copied from the installation media to the hard disk, and a GUI portion, where the installation is customized and completed.

8.2.1.1 Text portion of Windows 2000 installation

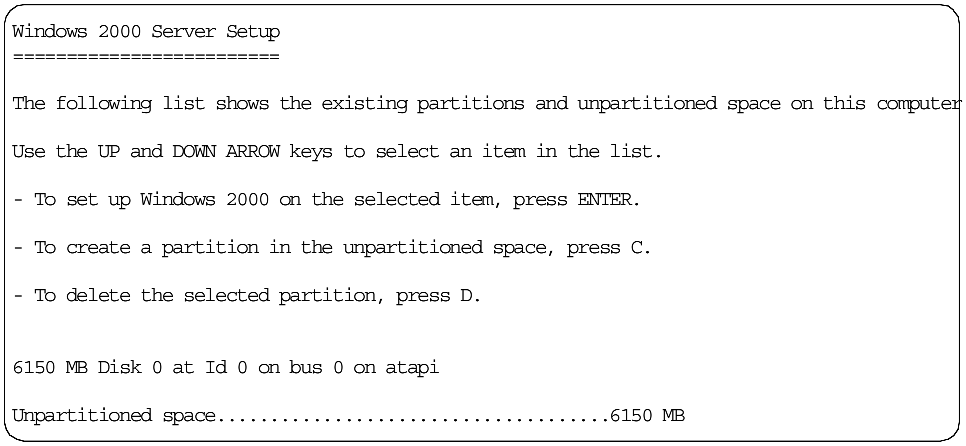
A system administrator can boot the system from CD-ROM if the BIOS of the PC supports it or from the accompanying floppy disks (four of them).

After the boot, the following screen is displayed:



```
Windows 2000 Server Setup
=========================

Welcome to Setup.
This portion of the Setup program prepares Microsoft(R)
Windows 2000(TM) to run on your computer.

- To set up Windows 2000 now, press ENTER.

- To repair a Windows 2000 installation, press R.

To quit Setup without installing Windows 2000, press F3.








ENTER=Continue R=Repair F3=Quit
```

After the license agreement screen, the system administrator has to format the system partition on which Windows 2000 will be installed:



```
Windows 2000 Server Setup
=========================

The following list shows the existing partitions and unpartitioned space on this computer

Use the UP and DOWN ARROW keys to select an item in the list.

- To set up Windows 2000 on the selected item, press ENTER.

- To create a partition in the unpartitioned space, press C.

- To delete the selected partition, press D.


6150 MB Disk 0 at Id 0 on bus 0 on atapi

Unpartitioned space.....................................6150 MB
```

If the partition is not already created, as in our example, the system administrator will have to define it. It is recommended that this partition should be at least 950 MB in size, but 2 to 4 GB is preferable.

After creating the partition, we must format it. It is strongly recommended to choose the NTFS file system since this is the only one that supports security.

After formatting the primary partition, Windows 2000 installation files are copied to installation folders (directories), and the system automatically reboots.

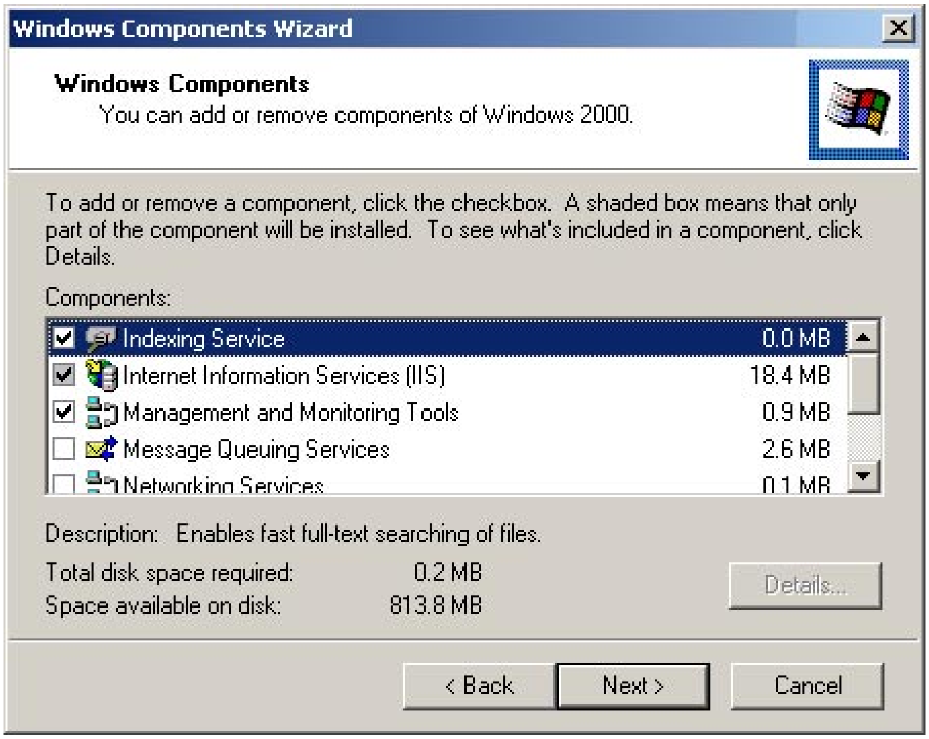
8.2.1.2 GUI portion of Windows 2000 Installation

After booting from the hard drive for the first time, the installation continues, and the user interface is now a GUI. All necessary files for basic Windows 2000 Server functionality are copied to the disk.

At this point, the system administrator can configure system locales (the way currency symbols, time, and date are represented) and keyboard layout. See section 8.2.10, “Windows 2000 support for national languages” on page 326, for details about which locales and languages are supported in Windows 2000.

The system administrator must then enter the name and the organization for which the Windows 2000 is licensed. Licences can be *Per Server* or *Per Seat*.

The computer name and the password for Administrator (similar to root user on AIX) must be entered on the next screen, and then the window with the list of Windows 2000 software components to be installed appears as shown in Figure 94 on page 287. If the system administrator chooses not to install any additional component at this time, he or she can always install it later. If the computer is connected to the network with the existing Microsoft Domain, the name of the Domain can be entered, and the system will be configured to that Domain automatically (if the Network Administrator added its name to the Domain before).



Windows Components Wizard

Windows Components
You can add or remove components of Windows 2000.

To add or remove a component, click the checkbox. A shaded box means that only part of the component will be installed. To see what's included in a component, click Details.

Components:

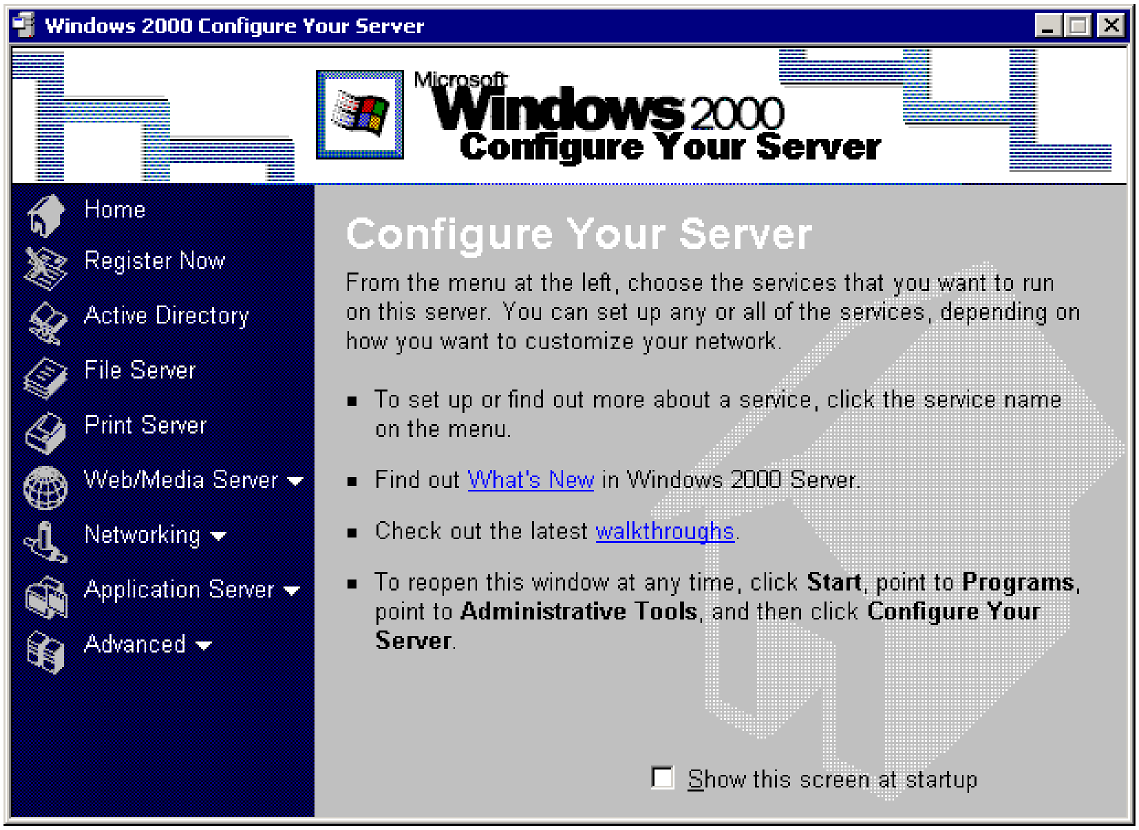
☑	Indexing Service	0.0 MB
☑	Internet Information Services (IIS)	18.4 MB
☑	Management and Monitoring Tools	0.9 MB
□	Message Queuing Services	2.6 MB
□	Networking Services	0.1 MB

Description: Enables fast full-text searching of files.

Total disk space required: 0.2 MB
Space available on disk: 813.8 MB

Figure 94. Windows 2000 installation - Additional components installation

After installing any additional component, the system reboots for the last time, and the utility for final configuration of the Server is started (see Figure 95 on page 288).



Configure Your Server

From the menu at the left, choose the services that you want to run on this server. You can set up any or all of the services, depending on how you want to customize your network.

- To set up or find out more about a service, click the service name on the menu.
- Find out What's New in Windows 2000 Server.
- Check out the latest walkthroughs.
- To reopen this window at any time, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Configure Your Server**.

☐ Show this screen at startup

Figure 95. Windows 2000 installation - Final configuration of the server

The system administrator can now perform all basic system configuration tasks, such as connecting to the network and configuring file and print servers, Web servers, or application servers. The detailed description of these tasks are well beyond the scope of this book. Refer to the Windows 2000 Server documentation for details.

8.2.1.3 Windows 2000 remote installation

It is also possible to install or upgrade a Windows 2000 system over the network. There are two possibilities:

- We already have the system installed with a previous version of the Windows operating system. In this case, installation can be started by mapping a network drive containing the Windows 2000 product and starting the winnt32.exe installation program. The installation must be started from the client system.

- Our existing client systems are not installed or we wish to make a fresh install instead of migrating old configurations. In this case, we can use Windows 2000's RIS (Remote Installation Services). We add the RIS component to our Windows 2000 Server and configure it to suit our needs regarding IS languages and other settings. If this is the only server in our environment, we must configure the DNS Server Service and the DHCP Server Service. The next step is to authorize the RIS server with the Active Directory. Then, we specify the directory on the server where the installation images for Windows 2000 are located and specify which client would be able to install Windows 2000 from this server (this is based on computer names, similar to the older NetBIOS names). In order to boot from the RIS Server, the clients must support DHCP PXE-based remote boot or a network adapter card supported by remote installation boot floppy. After the successful booting over the network, the user on the client is prompted with the Client Installation Wizard, which guides him or her through the Windows 2000 installation.

8.2.2 Windows 2000 boot process

The following is a description of the Windows 2000 boot process.

8.2.2.1 Preboot and boot sequences

The boot process is made up of a preboot sequence and a boot sequence. The preboot sequence comprises the following steps:

- Power-On Self Tests (POST) are run.
- The boot device is found, the Master Boot Record (MBR) is loaded into memory, and its program is run.
- The active partition is located, and the boot sector is loaded.
- The Windows 2000 loader (NTLDR) is then loaded.

The boot sequence executes the following:

- The Windows 2000 loader switches the processor to the 32-bit flat memory model.
- The Windows 2000 loader starts a mini file system.
- The Windows 2000 loader reads the BOOT.INI file and displays the operating system selections (boot loader menu).
- The Windows 2000 loader loads the operating system selected by the user. If Windows 2000 is selected, NTLDR runs NTDETECT.COM. For other operating systems, NTLDR loads BOOTSECT.DOS and gives it control.

- NTDETECT.COM scans the hardware installed in the computer and reports the list to NTLDR for inclusion in the Registry under the HKEY_LOCAL_MACHINE_HARDWARE hive.
- NTLDR then loads the NTOSKRNL.EXE and gives it the hardware information collected by NTDETECT.COM and enters the Windows 2000 load phases.

8.2.2.2 Windows 2000 load phases

When the Windows 2000 loader gives control to the Windows 2000 kernel, the Windows 2000 load phases are started. These phases are the kernel load phase, the kernel initialization phase, the services load phase, and the Windows subsystem start phase.

Kernel load phase

The Hardware Abstraction Layer (HAL) is loaded, and then the system hive is loaded and scanned for device driver services that should be loaded at this step.

Kernel initialization phase

This phase initializes the kernel and the drivers that were loaded in the previous phase. The system hive is again scanned to determine which high-level drivers should be loaded. These drivers are then initialized and loaded once the kernel has been initialized. The Registry hardware list is then created by using the information collected by NTDETECT.COM (on Intel-based systems) and OSLOADER.EXE (on RISC systems).

Services Load Phases

This phase starts the session manager (SMSS.EXE), which reads the list of programs that must be started. Usually, programs such as CHKDSK are executed at this step. Then, the paging file is set up, and the Win32 subsystem is started.

Windows subsystem start phase

When the Win32 subsystem starts, it automatically starts WINLOGON.EXE, which starts the Local Security Authority (LSASS.EXE) and displays the Ctrl+Alt+Del logon dialog box. Then, the Service Controller (SCREG.EXE) is run. It goes through the Registry and looks for services that must be loaded automatically. The boot is considered finished when a user can log on.

8.2.2.3 Booting in Windows 2000 Safe Mode

A new method that enables the system to boot in *Safe Mode* is added in Windows 2000. In this sort of boot, only a minimum number of device drivers are loaded. This is useful when a recently installed driver prevents the system from booting.

To enter safe mode, press F8 when Windows 2000 starts to boot. There are four different safe mode variations:

- **Standard safe mode** - This contains only the minimum number of drivers necessary for booting.
- **Networking-enabled** - Unlike standard mode, this also contains the drivers that enable networking.
- **safe mode with command prompt** - This is identical to standard mode except that the installation program runs the command prompt applications instead of Windows Explorer as the shell when the system is in the GUI installation phase.
- **Directory services repair mode** - This is valid only for Windows 2000 Domain Controllers and enables restoration of the Active Directory of a Domain Controller from backup media. In this mode, all of the device drivers are loaded; so, you cannot use it if one of the drivers prevents booting of the system; it is only for Directory services repair.

8.2.3 Windows 2000 configuration management

This section describes Windows configuration management mechanisms. Windows 2000 uses the Registry.

8.2.3.1 Windows 2000 Registry

The Windows 2000 Registry is a hierarchical centralized database containing configuration information, such as device driver settings, startup services, initialization files, and so on. The administrator adds, changes, and removes information in the registry indirectly by using tools from the Control Panel and Administrative Tools (see section 8.2.4.1, “Windows 2000 Control Panel” on page 296 and section 8.2.4.2, “Windows 2000 Administrative Tools” on page 301). There are also tools called Registry Editors (regedit.exe and regedt32.exe) that allow the system administrator to access the registry data directly if necessary.

In most cases, system changes to registry values can be accomplished by other means, but there are some settings that can only be altered via the Registry Editor; a Windows 2000 expert must have some knowledge about dealing directly with the Registry.

The Registry editors can be started by Start/Run regedit.exe or regedt32.exe.

Registry information

The registry divides all information about a computer and its users into several subtrees (see Figure 96 on page 293):

HKEY_LOCAL_MACHINE

This subtree is where most system configuration information is stored, such as information about hardware and software currently installed on the machine.

HKEY_CLASSES_ROOT

This subtree stores file association information (for example, a text file invoking Notepad) and information used by OLE (Object Linking and Embedding) technologies.

HKEY_USERS

This subtree contains user information. Default user and current user (listed by their security ID number) information is kept here.

HKEY_CURRENT_USER

This is the user profile of the user currently logged on interactively (not remotely).

HKEY_CURRENT_CONFIG

Information about the hardware profile used by the local computer at system startup.

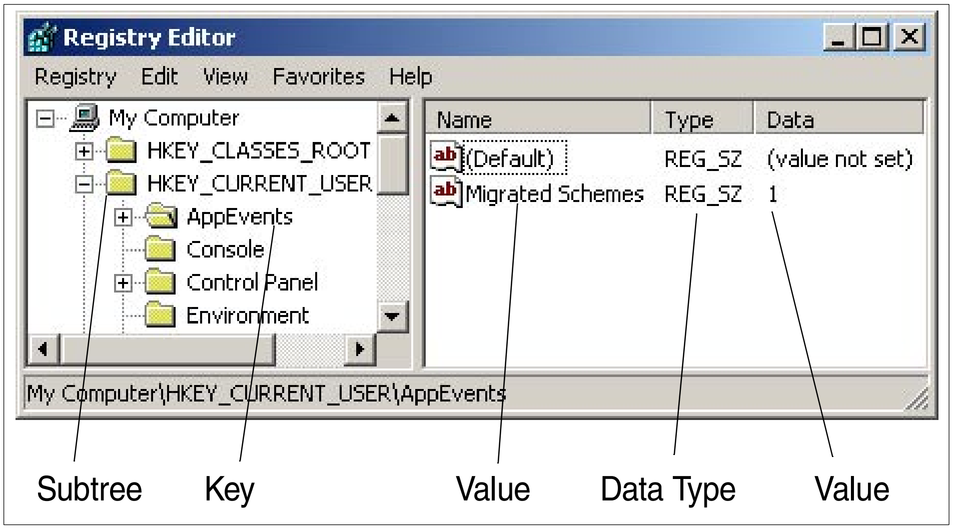


Figure 96. Windows 2000 Registry editor terminology

Keys are names of subsections of subtrees. They, in turn, contain other keys and/or values along with the values' data types. They can be hierarchically nested, much like a disk drive file system directory structure. As a matter of fact, Microsoft's naming convention has keys separated by backslashes as if they were directories or files in a file system. To use the file system analogy, subtrees are root drives; keys are directories, subdirectories, or files, which, in turn, contain the data (data type and value).

Just as the name implies, data type specifies what type of data is in the value for a given key. The data types are:

- **REG_BINARY** - Raw binary data
- **REG_DWORD** - Four bytes
- **REG_EXPAND_SZ** - Character string that can be expanded
- **REG_MULTI_SZ** - Multiple strings separated by nulls
- **REG_SZ** - String

The registry physically consists of several files associated with hives. A hive is a body of keys, subkeys and values that is rooted at the top of the registry hierarchy. These files reside in the *%SystemRoot%*\System32\Config folder.

The registry has built-in fault tolerance because every hive file has a file with the same name, but with a.LOG extension. Whenever a hive file is changed, the change is first written into its log file. In this way, the log file is not actually a backup file, but a journal of changes to the primary file. Once the description of the change to the hive file is complete, the log file is written immediately to disk, bypassing the disk cache. Once the file has successfully written to the disk, the system makes the appropriate changes to the hive file. If the system were to crash during the hive write operation, there is enough information in the journal file to roll back the hive to its previous position. The one exception to this procedure is with the SYSTEM hive. The SYSTEM hive contains very important information that needs to be fully backed up. If the original file is damaged, the system can use the backup to boot.

Registry editing tools

There are two tools used to interact with Windows 2000's Registry information. Because the administrator does not usually use these tools to change system configuration, these tools are not located on Windows 2000's Start menu. If you want to use them, you need to execute them from the Run menu, which is part of the Start menu, or from the command prompt.

`regedit.exe`

This program has the same interface as Windows 95's Registry Editor. The interface is also similar to Windows Explorer. This program provides a search function so that you can easily find the registry information you need.

`regedt32.exe`

This program provides some functions that regedit.exe does not have. For example, the program can load individual hives and set user security permissions and audit to each registry information. It is similar to the previous Windows File Manager. However, File Manager (and Explorer) can set the security permissions to files only on NTFS; regedt32.exe can set permissions on FAT as well as on NTFS.

8.2.3.2 Windows 2000 Device Configuration Management

During installation, Windows 2000 detects devices that are installed in the system and installs the appropriate device driver. However, not all devices are detected. For example, some network adapter cards need to be manually added to the system during installation setup so that the device driver (if available on the Windows 2000 CD-ROM) is installed.

At boot time, all devices are configured, and the appropriate device drivers are loaded to the system.

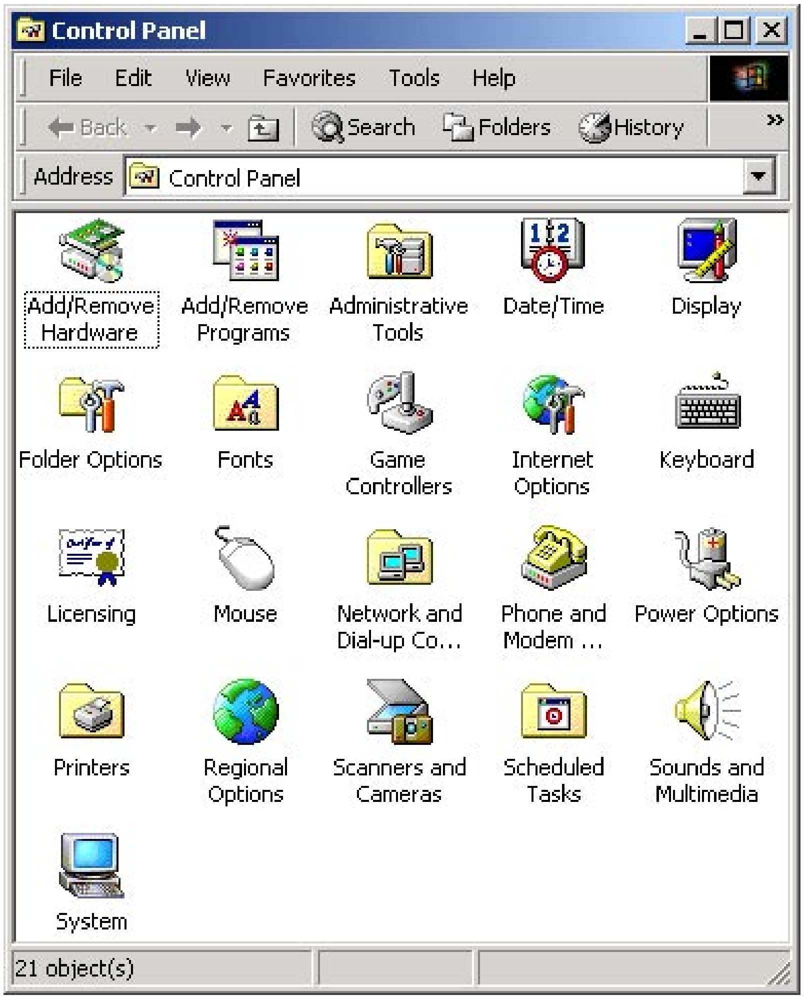


Figure 97. Default Windows 2000 Control Panel

Once the system is running, devices can be managed using specific icons available in the Control Panel, usually one per type of device. Devices cannot be controlled by using the Windows 2000 command prompt.

Figure 97 shows the Control Panel after a standard Windows 2000 Server installation:

For a description of the functions associated with each icon, see section 8.2.4.1, “Windows 2000 Control Panel” on page 296. Adding or removing a device or changing the status of a device can be handled through the appropriate icon.

8.2.4 Windows 2000 System Administration Interface

The Windows 2000 system administration interface is primarily located in the Control Panel and in the Administrative Tools folder.

The Control Panel is the main interface for device management as well as for controlling other aspects of the system. Non-administrative users can change some settings in the control panel as well. Administrative tools, as the name implies, provides tools for the administrator to monitor system events, manage hardware devices, users, Services, licensing, security policies, and some other tasks.

The number of tools that appear in these two interfaces depends on what kind of network module, application, or services have been installed; so, you may not find the following tools on your desktop, or you may find other modules that are not described in this document. The new tool for system management, called Microsoft Management Console was added to Windows 2000 (see section 8.2.4.3, “Windows 2000 Microsoft Management Console” on page 310), which enables the system administrator to manage local or remote systems.

8.2.4.1 Windows 2000 Control Panel

The Control Panel looks very similar to its Windows 3.x, Windows 9x, and Windows NT predecessors (see Figure 98 on page 297). However, there are some new functions that have been added to Windows 2000’s Control Panel. All of the Control Panel settings are stored in a centralized hierarchical database called the Registry, which is described in section 8.2.3.1, “Windows 2000 Registry” on page 291. This database is similar to the AIX ODM. While any values that can be changed in the Control Panel can be changed directly using the Registry Editor (regedit.exe), the Control Panel provides an intuitive interface and serves to protect the user from incorrectly changing registry values.

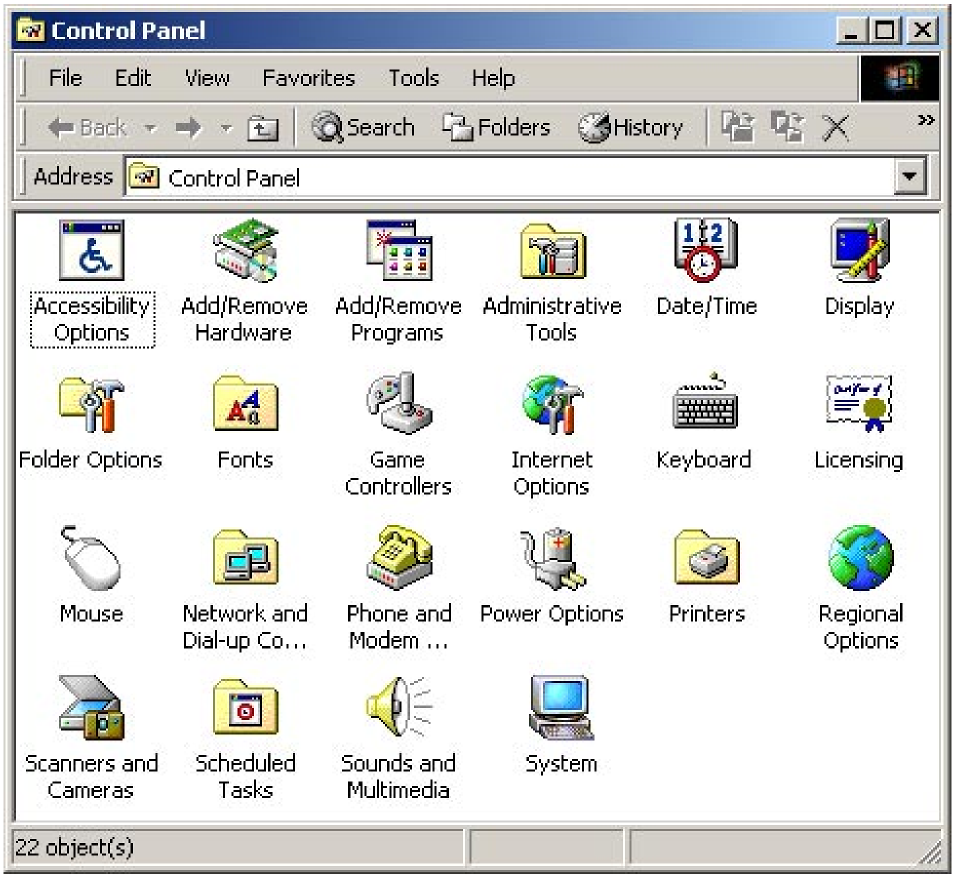


Figure 98. Windows 2000 Control Panel

Items that can be controlled in the Control Panel range from setting desktop color to configuring networking adapters and protocols. Many of the Control Panel settings are stored on a per-user basis so that each user can customize their own desktop. There are some settings, for example Network Settings, that are stored on a per-machine basis. Each Control Panel icon invokes an applet to control a particular group of settings. Brief descriptions of the Control Panel applets and a discussion of the features they offer follow.

Accessibility Options

These options help people with disabilities manage their keyboard, mouse, and visual and audio outputs of their system. For example, enabling a tone whenever you press the key, enabling the pressing of one key at a time for sequences such as Ctrl+Alt+Del (StickyKeys), generating visual warnings when a sound is emitted, using colors and fonts for easy reading, and so on.

Add/Remove Programs

This controls installation and uninstallation for Windows 2000 applications and Windows 2000 setup. You can easily add or remove an application or a Windows 2000 component.

Add/Remove Hardware

This starts the Add/Remove Hardware Wizard, which lets you install, unplug, troubleshoot, or remove hardware components.

Administrative Tools

This opens a new window with options which are described in section 8.2.4.2, “Windows 2000 Administrative Tools” on page 301.

Date/Time

This allows the setting of date and time. Also, the time zone is selected here, and an automatic adjustment for daylight savings time can be enabled or disabled.

Display

The display applet controls the display drivers, along with color palette depth, screen resolution, font size, and screen refresh frequency. After the display settings are chosen, there is a test button that will display a test pattern with the new settings for five seconds.

Windows 2000 is sensitive to display changes. It is very important that before the system administrator allows Windows 2000 to shut down, the test be passed, or Windows 2000 will restart with no video. To circumvent this problem, there is an option at startup called *Last known good*, which will restore the previous driver information so that the system can be restarted.

You can also change the system color scheme and adjust the color palette here. These settings are stored on a per-user basis.

Folder Options

The configuration of the appearance of files in folders and of folders themselves can be set in this applet. The system administrator can also enable the appearance of Web contents on the desktop and/or in the folders. He or she can set the association of filename extensions with applications.

Fonts

This manages installed Windows 2000 fonts. It allows the installation and removal of TrueType, OpenType (such as Adobe type 1 fonts), Raster fonts, and Vector fonts. This information is stored on a machine-wide basis. Fonts are available to both 16-bit and 32-bit Windows applications.

Game Controllers

Joysticks and similar devices can be configured here.

Internet Options

The system administrator can configure and customize Microsoft Internet Explorer; for example, the user can define the URL of the home page, customize security settings, and configure dial-up connections and mail programs.

Keyboard

This allows fine-tuning of keyboard key repeat rate and delay before key repeat.

Licensing

This manipulates the Licensing Mode. The administrator selects the proper mode from Per Server (concurrent mode) and Per Seat (fixed license on one machine).

Mouse

Mouse preferences, such as tracking speed and double-click speed, are set here. There are also options for left- and right-handed users. There is no system support for three-button mouses, but some applications may support the third button with additional software.

Cursors, such as arrows, busy pointers, application pointers, and drag pointers, can also be selected here. There is a selection of static and animated pointers included with Windows 2000.

Network and Dial-up Connections

The system administrator can manage existing network adapters and protocols or use the Wizard to create new ones.

Phone and Modem

Configures phone dialing and modem setting.

Power Options

This controls the configuration of power saving modes and the Uninterruptable Power Supply (UPS). If the system is attached to UPS and UPS is connected to one of the serial ports, it can be configured to notify the system administrator or to gracefully shut down the system (after a specified amount of time) in case of power failure.

Printers

The system administrator can manage existing printers or use a Wizard to add new printers to the system.

Regional Options
This allows control and configuration of country, language, and keyboard layout. These settings are per-machine, not per-user. NLS versions of Windows 2000 provide complete non-English language solutions. You can find more information on localization of Windows 2000 in section 8.2.10, “Windows 2000 support for national languages” on page 326.

Scanners and Cameras
The system administrator can manage the properties of installed scanners and video cameras or use a Wizard to add new ones.

Scheduled Tasks
The system administrator can configure a task (start of a script or an application) to be performed at a later time.

Sounds and Multimedia
The system administrator can configure different sounds that can be played when certain events occur on the machine. It is also possible to configure installed audio hardware.

System
The system administrator can change various system properties like adding new hardware or start the Device Manager to configure existing devices as seen in Figure 99 on page 301.

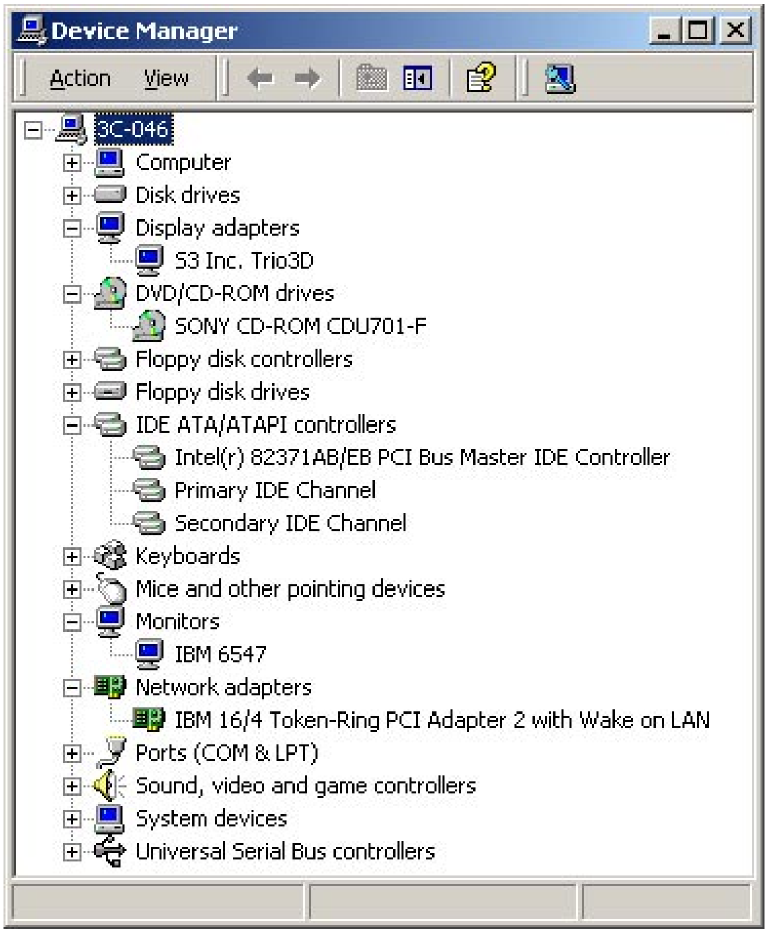


Figure 99. Windows 2000 Device Manager

8.2.4.2 Windows 2000 Administrative Tools

The following applets are included in the Administrative Tools (see Figure 100 on page 302) to provide an interface for the administrator to monitor and control the local system and/or other systems on the network. Note that the number of applets will depend on the number of installed software components, and you may not see exactly the same window content on your system.

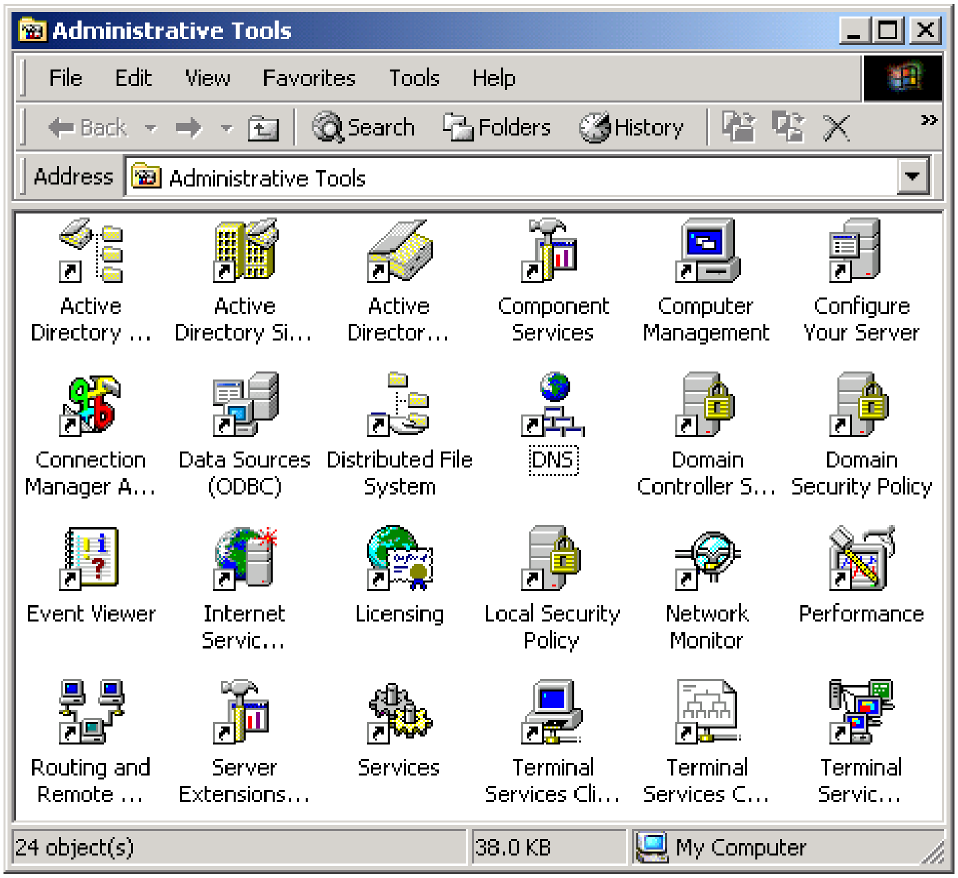


Figure 100. Windows 2000 Administrative Tools

Active Directory Domains and Trusts

This helps the administrator manage trust relationships between domains.

Using Active Directory Domains and Trusts, you can:

- Provide interoperability with other domains, such as pre-Windows 2000 domains or domains in other Windows 2000 forests, by managing trusts.
- Change the mode of operation of a Windows 2000 domain from mixed-mode to native-mode.
- Add and remove alternate UPN suffixes used to create user logon names.
- Transfer the domain naming operations master role from one domain controller to another.

Active Directory Sites and Services

The primary purpose of Active Directory Sites and Services is to administer the replication topology both within a site in a local area network and between sites in a wide area network in your enterprise environment.

Active Directory Users and Computers

This allows you to add, move, delete, and alter the properties for objects such as users, contacts, groups, servers, printers, and shared folders.

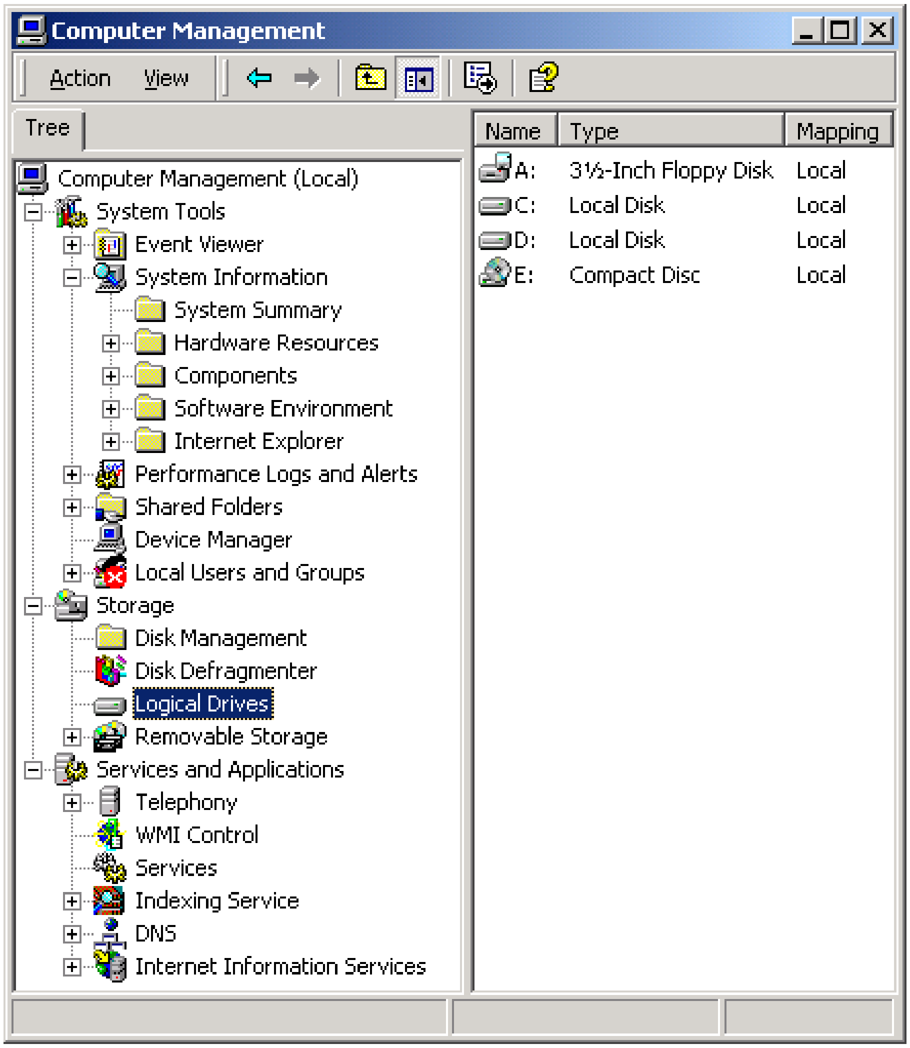
Component Services

This section covers the configuration and management of COM+ applications. COM application is a term describing a group of components developed to work together, such as Microsoft Word and Microsoft Excel. These applications consist of executable files and several DLLs (application extensions) for additional functionality.

COM+ applications are a group of COM components developed and configured together with additional functionality, such as queueing and role-based security.

Computer management

This is the primary tool for the management of disks, logical drives, remote storage, users, and groups (see Figure 101 on page 304). There are also links to other tools for managing local and remote systems, such as Event Viewer, Performance Logs, Device Manager, and so on, which will be covered later in this section.



Name	Type	Mapping
A:	3½-Inch Floppy Disk	Local
C:	Local Disk	Local
D:	Local Disk	Local
E:	Compact Disc	Local

Figure 101. Windows 2000 Computer Management

Configure Your Server

This is the tool that is started automatically after a new Windows 2000 Server installation. The system administrator can also start it at a later time to perform tasks, such as configuring Active directory, Networking (DHCP, DNS, Remote Access, Routing), and configuring the system to act as a File server, Print server, Web/Media server, and Application server. These tasks can also be accessed directly from the Control Panel or Administrative Tools menu.

Connection Manager Administration Kit
A Wizard guides the system administrator through tasks needed to configure dial-up connections for users of the system, such as support for Virtual Private Networks (VPN).

Data Sources ODBC
Data Source Open Database Connectivity (ODBC) is used to access the data from other database management systems, such as SQL databases and Visual FoxPro database.

Distributed File System (DFS)
This is a tool for managing DFS, enabling you to organize logical shared disks on multiple systems to act as one logical structure. For more details about DFS, see section 6.2.2.11, "Distributed File System (DFS)" on page 149.

Domain Name System (DNS)
This tool is used for configuring and managing DNS, which is a hierarchical naming system for computer systems connected to an IP network. For more information about DNS, see Chapter 10, "Networking" on page 361.

Domain Controller Security Policy
Domain Controller Security Policy manages security policy for domain controllers.

Domain Security Policy
Domain Security Policy manages security policy for the domain, including updating user rights and audit policies. These included:

- User Password, or Account Policy to control how passwords are used by user accounts.
- Audit Policy to control what types of events are recorded in the security log.
- User Rights are applied to groups or users, and effect the activities permitted on an individual workstation, a member server, or on all domain controllers in a domain.

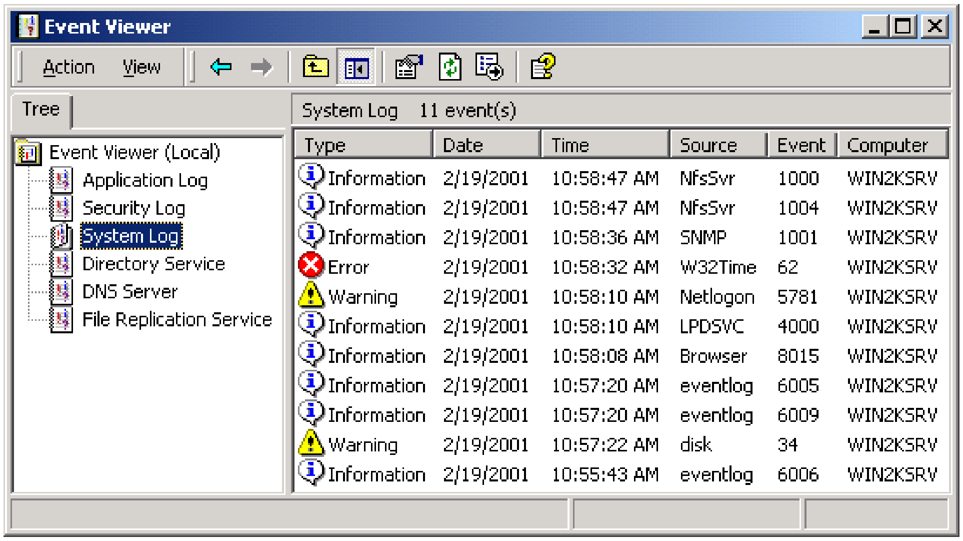
Event Viewer
Event Viewer can be used to monitor system event logs, divided into three default categories:

Application log Events logged by user applications.

Security log Events about valid and invalid login attempts and resource use.

System log Events logged by system components, such as device drivers.

The tool is shown in Figure 102 on page 306.

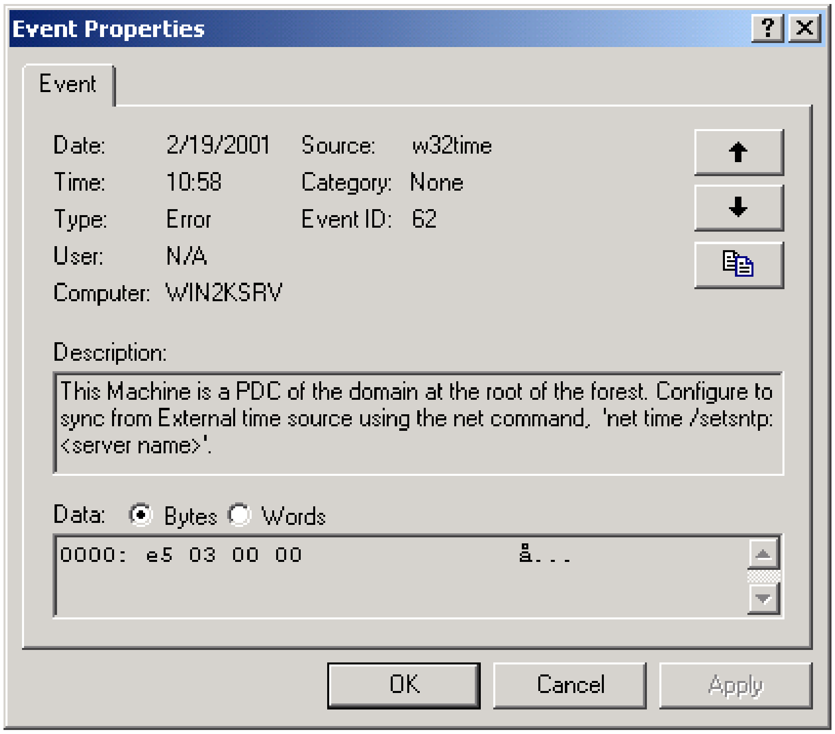


Type	Date	Time	Source	Event	Computer
Information	2/19/2001	10:58:47 AM	NfsSvr	1000	WIN2KSRV
Information	2/19/2001	10:58:47 AM	NfsSvr	1004	WIN2KSRV
Information	2/19/2001	10:58:36 AM	SNMP	1001	WIN2KSRV
Error	2/19/2001	10:58:32 AM	W32Time	62	WIN2KSRV
Warning	2/19/2001	10:58:10 AM	Netlogon	5781	WIN2KSRV
Information	2/19/2001	10:58:10 AM	LPDSVC	4000	WIN2KSRV
Information	2/19/2001	10:58:08 AM	Browser	8015	WIN2KSRV
Information	2/19/2001	10:57:20 AM	eventlog	6005	WIN2KSRV
Information	2/19/2001	10:57:20 AM	eventlog	6009	WIN2KSRV
Warning	2/19/2001	10:57:22 AM	disk	34	WIN2KSRV
Information	2/19/2001	10:55:43 AM	eventlog	6006	WIN2KSRV

Figure 102. Windows 2000 Event Viewer

Since this server is a Windows 2000 Domain Controller with a locally installed DNS, we have three additional Event Viewer categories: Directory Service, DNS Server, and File Replication Service.

If the particular event is double-clicked with the mouse, in this case the error message generated by W32Time, the event properties are displayed with detailed information about the event and, sometimes, the possible corrective actions needed (see Figure 103 on page 307).



Date: 2/19/2001 Source: w32time
Time: 10:58 Category: None
Type: Error Event ID: 62
User: N/A
Computer: WIN2KSRV

Description:

This Machine is a PDC of the domain at the root of the forest. Configure to sync from External time source using the net command. 'net time /setsntp:<server name>'.

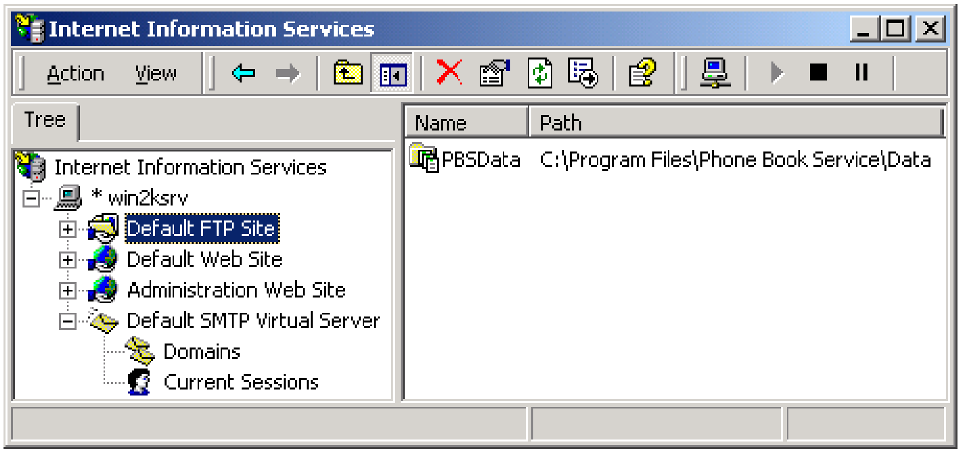
Data: ● Bytes ○ Words

```
0000: e5 03 00 00                         å...
```

Figure 103. Windows 2000 Event Viewer - Event Properties

Internet Services Manager

With this tool, system administrators can manage the various IIS components such as the Web server, FTP server, and SMTP server. See Figure 104 on page 308 to see an example of this interface.



Name	Path
PBSData	C:\Program Files\Phone Book Service\Data

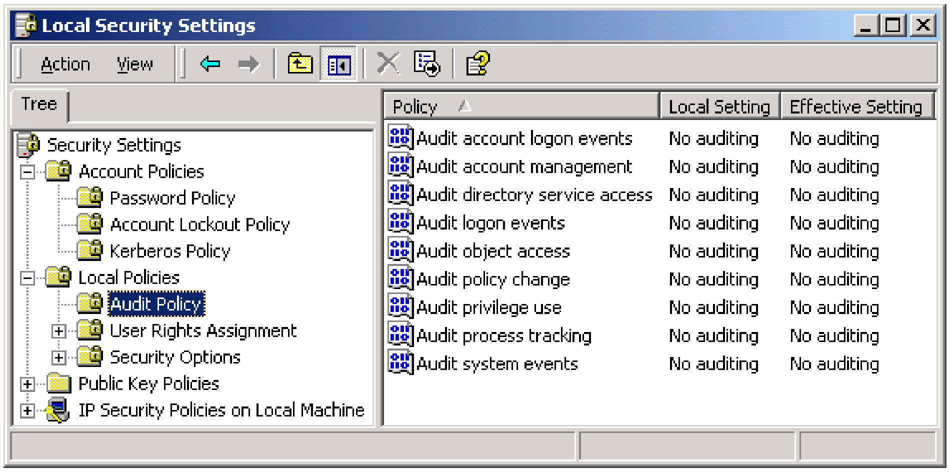
Figure 104. Windows 2000 Internet Services Management

Licensing

This tool enables license management for Windows servers on your network.

Local Security Policy

With this tool, the system administrator can modify local user rights, permissions, and auditing policies (see Figure 105).

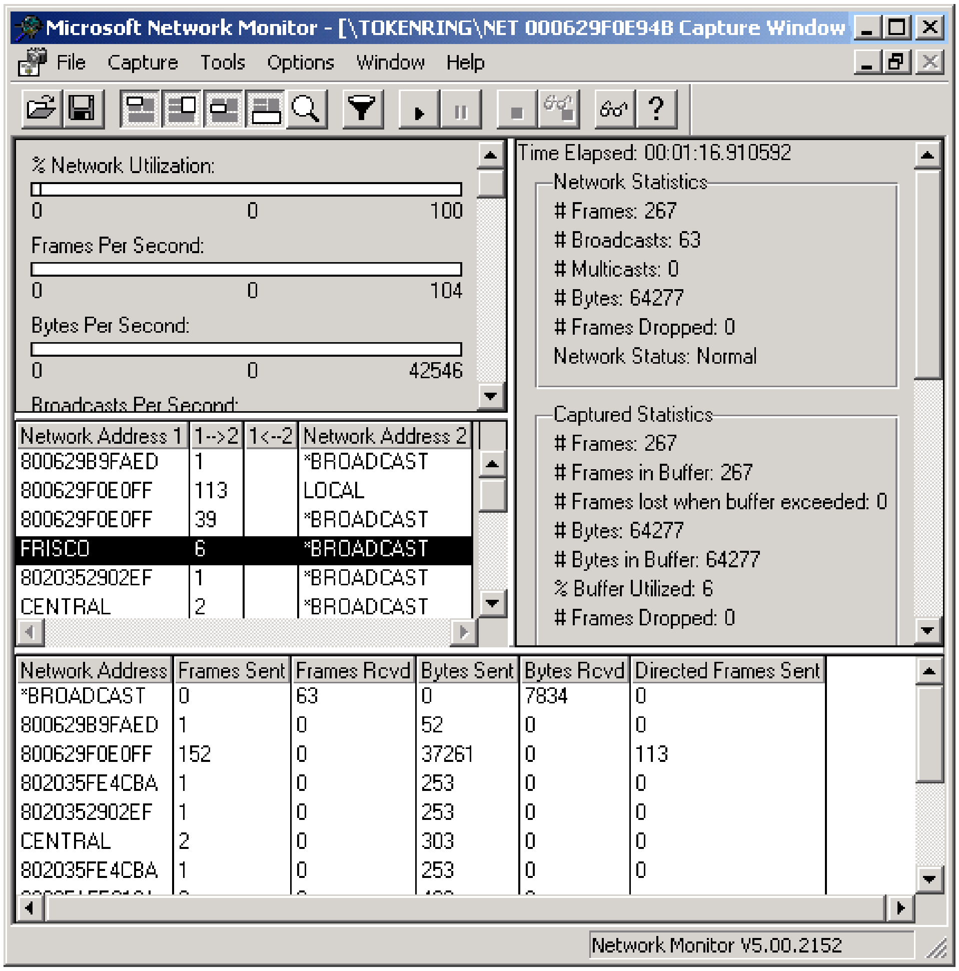


Policy	Local Setting	Effective Setting
Audit account logon events	No auditing	No auditing
Audit account management	No auditing	No auditing
Audit directory service access	No auditing	No auditing
Audit logon events	No auditing	No auditing
Audit object access	No auditing	No auditing
Audit policy change	No auditing	No auditing
Audit privilege use	No auditing	No auditing
Audit process tracking	No auditing	No auditing
Audit system events	No auditing	No auditing

Figure 105. Windows 2000 Local Security Policy tool

Network Monitor

This tool enables the system administrator to capture and analyze network traffic and aids in detecting network problems (see Figure 106 on page 309).


Microsoft Network Monitor - [\TOKENRING\NET 000629F0E94B Capture Window
File
Capture
Tools
Options
Window
Help
Network Monitor V5.00.2152

% Network Utilization:
0 0 100
Frames Per Second:
0 0 104
Bytes Per Second:
0 0 42546
Broadcasts Per Second:

Time Elapsed: 00:01:16.910592

Network Statistics
Frames: 267
Broadcasts: 63
Multicasts: 0
Bytes: 64277
Frames Dropped: 0
Network Status: Normal

Captured Statistics
Frames: 267
Frames in Buffer: 267
Frames lost when buffer exceeded: 0
Bytes: 64277
Bytes in Buffer: 64277
% Buffer Utilized: 6
Frames Dropped: 0

Network Address 1	1-->2	1<--2	Network Address 2
800629B9FAED	1		*BROADCAST
800629F0E0FF	113		LOCAL
800629F0E0FF	39		*BROADCAST
FRISCO	6		*BROADCAST
8020352902EF	1		*BROADCAST
CENTRAL	2		*BROADCAST

Network Address	Frames Sent	Frames Rcvd	Bytes Sent	Bytes Rcvd	Directed Frames Sent
*BROADCAST	0	63	0	7834	0
800629B9FAED	1	0	52	0	0
800629F0E0FF	152	0	37261	0	113
802035FE4CBA	1	0	253	0	0
8020352902EF	1	0	253	0	0
CENTRAL	2	0	303	0	0
802035FE4CBA	1	0	253	0	0

Figure 106. Windows 2000 Network Monitor

Performance

This tool can be used for monitoring performance and resource utilization of the system. First, you must specify which resources you would like to monitor, and then start the capturing. After a set amount of time you turn off the monitoring and analyze the results (see Chapter 9, “Performance monitoring” on page 335).

Routing and Remote Access

The routing and remote access service provides multi-protocol routing, remote access, and VPN (Virtual Private Network) capabilities. This service can route IP, IPX or AppleTalk protocols using RIP, OSPF or SAP routing protocols over LANs or WANs. Access can be managed on a user- or on a domain basis.

Server Extension Administration

This tool helps Web authors include and manage the following functions:

- Collaboration on creating and maintaining Web sites.
- Support for hit counters, full-text searches, or e-mail form handling.
- Work on multiple server platforms (Windows 2000, UNIX) and on multiple Web servers (IIS, WebSite, Netscape).
- Integration with Microsoft Office, Visual SourceSafe, and Index Server files.

Services

This tool enables the system administrator to manage Windows 2000 Services. A Service is a program, routine, or process that performs a specific system function and usually supports some other program (very similar to daemons in UNIX operating systems). With this tool, the system administrator can display all services and can start, stop, pause, resume, restart or change properties, such as the type of start and the startup parameters of a particular service.

Terminal Services

Terminal Services provide access to a Windows 2000 Server from other clients on the network through *thin client* SW, similar to terminal emulation.

With the set of these three tools the system administrator can configure and manage the Windows 2000 Terminal Services:

- Terminal Services Client Creator
- Terminal Services Configuration
- Terminal Services Manager

If Terminal Services Licensing is installed a fourth tool is displayed.

For more details about Terminal Services, see section 8.2.9, “Windows 2000 Terminal services” on page 325.

8.2.4.3 Windows 2000 Microsoft Management Console

The new tool that helps the system administrator administer Windows 2000 is called the Microsoft Management Console (MMC). MMC is a framework with which the system administrator can create a collection of other administrative tools in so-called consoles. Each console can contain monitor controls, task controls, wizards, and documentation used to manage hardware, software, or networking components on a local or remote computer.

The system administrator can start an empty MMC by entering `mmc` in a Start/Run window.

8.2.4.4 Windows 2000 Remote Management

The system administrators have several choices for managing remote Windows 2000 systems:

- Managing a remote system through Terminal Services

  The system administrator can remotely log on and manage any Windows 2000 system on the network providing he or she has access permission. In order to effectively use this service, the system administrator should have at least a 28.8 Kbps or faster modem connection to the managed system. For more details about Terminal Services, see section 8.2.9, “Windows 2000 Terminal services” on page 325.

- Manage the remote systems through the Microsoft Management Console.

  When the system administrator configures their MMC, he or she can choose whether each tool will be used for a local or remote system. For more details about MMC, see section 8.2.4.3, “Windows 2000 Microsoft Management Console” on page 310.

- Mange the remote systems through Administration Tools

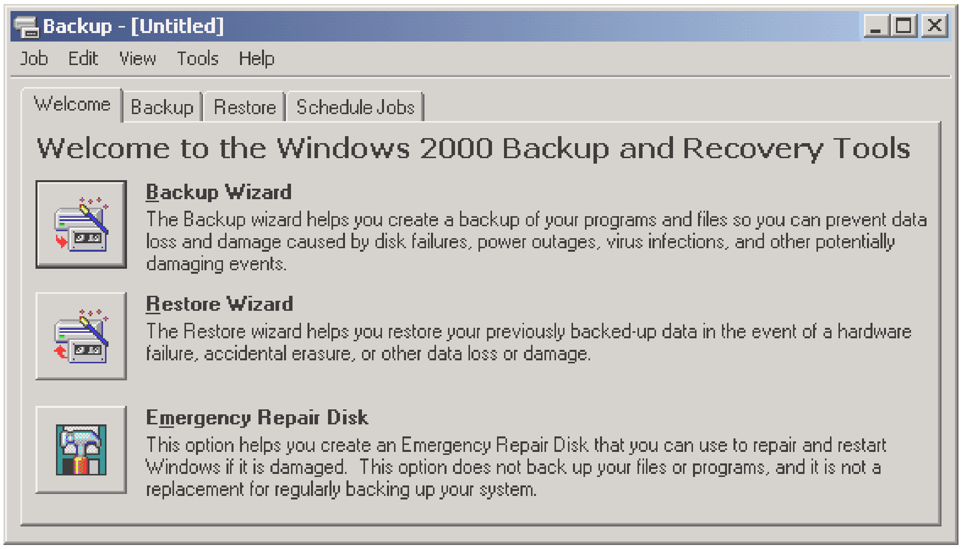
  Most of the tools in Administration Tools panel can be used for administration of the remote system since these essentially are customized MMC consoles. For details about the Administrative Tools folder, see section 8.2.4.2, “Windows 2000 Administrative Tools” on page 301.

- Scripting with Windows Scripting Host

  The system administrator can use Windows Scripting Host (WSH) and automate administrative actions on remote systems, such as creating shortcuts, mapping network drives, configuring printers, and so on. WSH is fairly language-independent, and scripts can be written in Visual Basic Scripting Edition, JScript, Perl, or a similar language.

8.2.5 Windows 2000 backup/restore

The system administrator can use the Backup tool to back up user and system files on the Windows 2000 System (see Figure 107 on page 312).



Welcome to the Windows 2000 Backup and Recovery Tools

Backup Wizard
The Backup wizard helps you create a backup of your programs and files so you can prevent data loss and damage caused by disk failures, power outages, virus infections, and other potentially damaging events.

Restore Wizard
The Restore wizard helps you restore your previously backed-up data in the event of a hardware failure, accidental erasure, or other data loss or damage.

Emergency Repair Disk
This option helps you create an Emergency Repair Disk that you can use to repair and restart Windows if it is damaged. This option does not back up your files or programs, and it is not a replacement for regularly backing up your system.

Figure 107. Windows 2000 Backup and Recovery Tools

The system administrator can use the *Backup Wizard* to back up the data or manually select items with the Backup tab. If the backup is to be made to a tape device, this must be connected and configured prior to starting the Backup Wizard.

The Backup tool is also where you create the Emergency Repair Disk for your system. In Windows NT 4.0, this was done using a separate utility called RDisk.exe.

When files from an NTFS file system are restored, make sure the destination file system is NTFS as well. This helps preserve permissions, encryption of files or folders, disk quota information, mounted drive information, and remote storage information.

Types of backups

There are five different types of backups in Windows 2000:

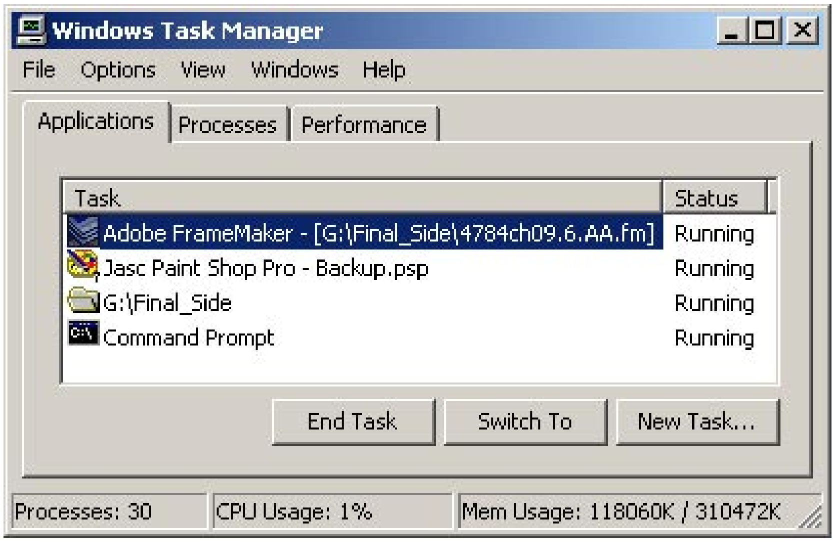
- **Normal Backup** - This archives all selected files and clears the *Archive attribute* of a file.
- **Incremental Backup** - This copies only the files that were changed since the last normal or incremental backup (only the files with the Archive attribute set) and clears their Archive attribute. If you want to restore the complete disk or folder, you must have the media with the first Normal backup (complete backup) and then all media with consecutive

Incremental backups. If you only want to install one file, it is sufficient to have the media on which this file was backed up last time. As can be seen, this procedure can be quite challenging to implement.

- **Differential Backup** - This is the same as incremental but does not clear the Archive attribute.
- **Copy Backup** - This is the same as Normal backup but does not clear the Archive attribute.
- **Daily Backup** - This copies all files that have changed during the day and does not clear the Archive attribute.

8.2.6 Windows 2000 Process Management

Processes are managed with the Task manager. The Task Manager can be invoked by simultaneously pressing Ctrl+Shift+Esc or by selecting Task Manager from the Windows 2000 security menu (accessed by pressing Ctrl+Alt+Del). Figure 108 shows the Task Manager Applications tab.



Task	Status
Adobe FrameMaker - [G:\Final_Side\4784ch09.6.AA.fm]	Running
Jasc Paint Shop Pro - Backup.psp	Running
G:\Final_Side	Running
Command Prompt	Running

Processes: 30 CPU Usage: 1% Mem Usage: 118060K / 310472K

Figure 108. Windows 2000 Task Manager - Applications tab

By right clicking on an application in the task list, it is possible to go to the corresponding process on the Processes tab or to end the application.

On the Processes tab, seen in Figure 109 on page 314, a list of all processes with some additional information is shown. The number of columns (essentially, what type of information) can be selected from the **View** -> **Select Columns** menu. The columns can be sorted either in an ascending or a descending fashion by clicking on the column headers. This is a very convenient way to find out which processes use the most system resources by sorting the CPU Time column.

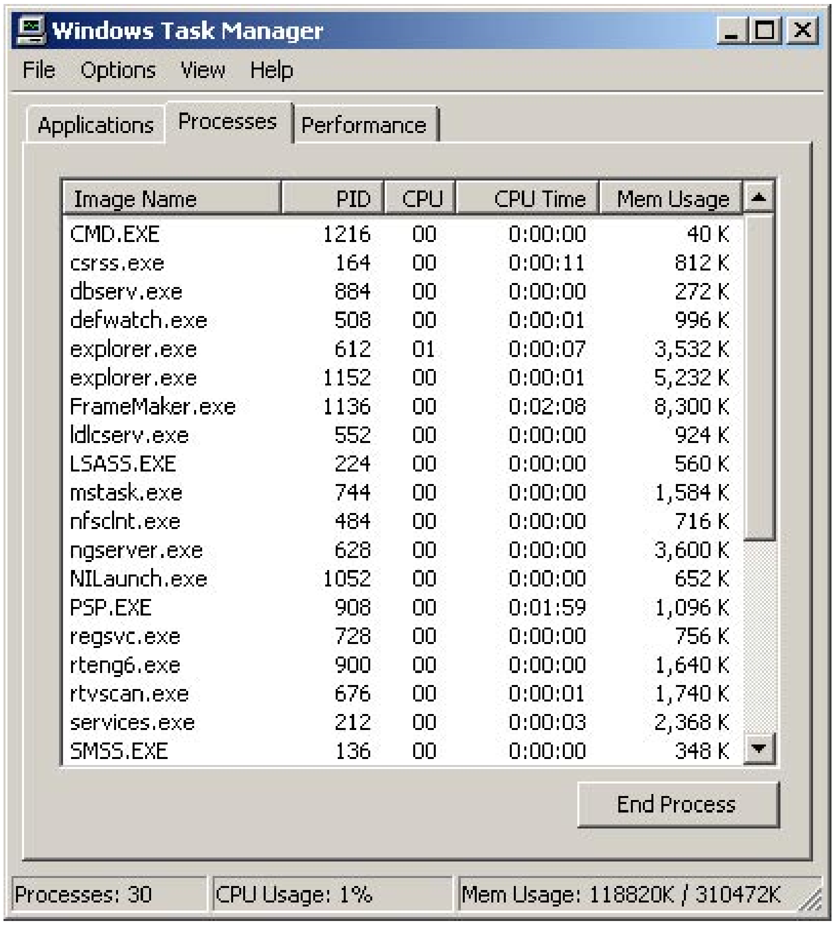


Image Name	PID	CPU	CPU Time	Mem Usage
CMD.EXE	1216	00	0:00:00	40 K
csrss.exe	164	00	0:00:11	812 K
dbserv.exe	884	00	0:00:00	272 K
defwatch.exe	508	00	0:00:01	996 K
explorer.exe	612	01	0:00:07	3,532 K
explorer.exe	1152	00	0:00:01	5,232 K
FrameMaker.exe	1136	00	0:02:08	8,300 K
ldlcserv.exe	552	00	0:00:00	924 K
LSASS.EXE	224	00	0:00:00	560 K
mstask.exe	744	00	0:00:00	1,584 K
nfsclnt.exe	484	00	0:00:00	716 K
ngserver.exe	628	00	0:00:00	3,600 K
NILaunch.exe	1052	00	0:00:00	652 K
PSP.EXE	908	00	0:01:59	1,096 K
regsvc.exe	728	00	0:00:00	756 K
rteng6.exe	900	00	0:00:00	1,640 K
rtvscan.exe	676	00	0:00:01	1,740 K
services.exe	212	00	0:00:03	2,368 K
SMSS.EXE	136	00	0:00:00	348 K

Processes: 30 | CPU Usage: 1% | Mem Usage: 118820K / 310472K

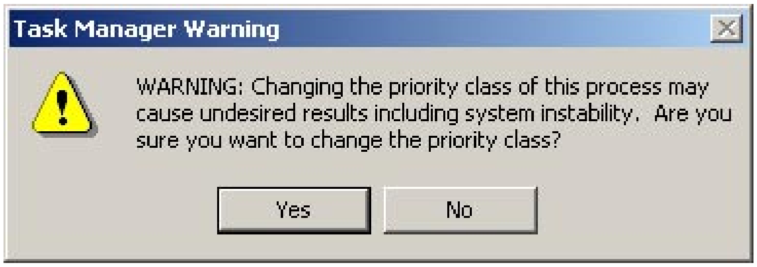
Figure 109. Windows 2000 Task Manager - Processes tab

Right clicking on a process allows you to end the process itself, the entire process tree in which the process exists, or set the priority of this process to any of the following levels:

- Realtime
- High
- Above Normal
- Normal (default)
- Below Normal

- Low

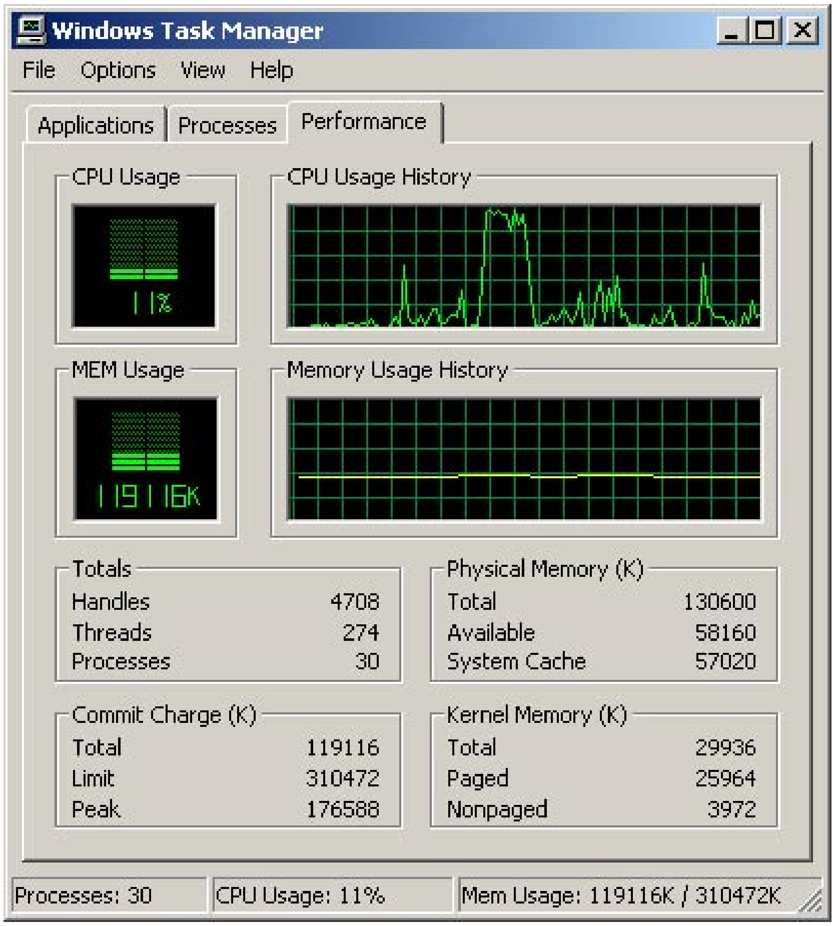
However, be careful when changing the priority of a process, especially if you assign any of the levels above Normal as this easily could make your system unresponsive. As a precaution, the message in Figure 110 will appear after all process changes to make sure you really want to do this.


Task Manager Warning

WARNING: Changing the priority class of this process may cause undesired results including system instability. Are you sure you want to change the priority class?

Figure 110. Windows 2000 Task Manager - Priority warning message

The last tab in the Task Manager, the Performance tab, is used to display system resource usage graphically over time, just like a small performance monitor applet if you will. An example of this is shown in Figure 111 on page 316.



Totals	
Handles	4708
Threads	274
Processes	30

Physical Memory (K)	
Total	130600
Available	58160
System Cache	57020

Commit Charge (K)	
Total	119116
Limit	310472
Peak	176588

Kernel Memory (K)	
Total	29936
Paged	25964
Nonpaged	3972

Processes: 30 | CPU Usage: 11% | Mem Usage: 119116K / 310472K

Figure 111. Windows 2000 Task Manager - Performance tab

8.2.7 Windows 2000 reliability, availability, and serviceability

Windows 2000 provides high availability in three ways: By uniformly handling hardware and software system faults, by protecting the system from user programs, and providing data and system recovery mechanisms. Windows 2000 includes a variety of reliability and fault-tolerance capabilities that will be covered in the following sections.

8.2.7.1 Operating system protection

In order to increase the reliability and availability of Windows 2000, Microsoft implemented or enhanced (compared to previous versions of Windows NT) the following concepts:

Protected subsystems

Windows 2000 employs protected subsystems in its design and forms a separation of user and kernel modes. This protects the Windows 2000 kernel from misbehaving applications (or any user mode program). Each protected subsystem resides in its own protected memory space, as does each Windows 2000 application. This architecture provides a stable environment.

Kernel-mode write protection

Windows 2000 *Memory Manager* provides write protection for parts of the kernel, such as kernel code, read-only subsections and device drivers. This protection is implemented in such a way that even parts of the kernel cannot overwrite other (protected) parts of the kernel.

Windows File Protection

Windows File Protection enables the protection of system files so that they cannot be replaced by application installations.

8.2.7.2 Recoverable File System

Windows 2000 also provides handling of hardware faults, such as disk and disk-related failures. Much of this disk fault tolerance is related to NTFS, the Windows 2000 file system. NTFS is a comprehensive recoverable file system that provides virtually instant recovery from a disk failure. The file system logs each disk I/O operation as a unique transaction. When a user updates a file, the Log File Service (LFS) logs redo and undo information for that transaction. Redo is the information that tells NTFS how to repeat the transaction. Undo tells NTFS how to roll back the transaction. If a transaction completes successfully, the file update is committed. If the transaction is incomplete, NTFS ends or rolls back the transaction by following the instructions in the undo information. If NTFS detects an error in the transaction, the transaction is also rolled back.

File system recovery is straightforward with NTFS. If the disk fails but is not destroyed (for example, due to a power outage), NTFS performs three passes: An analysis pass, a redo pass, and an undo pass. During the analysis pass, NTFS appraises the damage and determines exactly which clusters must now be updated per the information in the log file. The redo pass performs all transaction steps logged from the last checkpoint. The undo pass backs out any incomplete (uncommitted) transactions.

In addition to virtually instant recovery, NTFS supports hot-fixing. If an error occurs due to a bad sector, NTFS moves the information to a different sector and marks the original sector as bad. This process is completely transparent to an application performing disk I/O.

For more details about the NTFS file system, see section 6.2.2, “Windows 2000 file systems” on page 137.

8.2.7.3 Auto restart

In the event of a system failure, Windows 2000 can be configured to automatically restart itself. There is also an option to write a memory dump file to disk before restarting for later analysis. In systems with a lot of system memory (RAM) the Kernel dump can be quite a large structure (slightly larger than the amount of RAM). In this case, enabling the Small Kernel Dump option will only dump Kernel Information (Kernel Structures) to disk and not the whole amount of physical memory.

8.2.7.4 Tape backup/restore support

Tape backup is included with the Windows 2000 product. Refer to section 8.2.5, “Windows 2000 backup/restore” on page 311, for a more detailed description.

8.2.7.5 UPS support

An Uninterruptable Power Supply (UPS) is a battery-operated power supply connected to a computer that keeps the system running during a power failure. The UPS service for Windows 2000 detects and warns users of power failures and manages a safe system shutdown when the backup power supply is about to fail.

8.2.7.6 RAID support

Fault-tolerant disk systems are standardized and categorized in a number of levels. Each level offers various mixes of performance, reliability, and hardware cost. See section 6.2.3, “Windows 2000 RAID support” on page 149, for more information about RAID.

8.2.7.7 Diagnostic, recovery, and repair tools

The following diagnostic, recovery, and repair tools are available in Windows 2000.

System information

The system administrator can use system information to check for resource conflicts, hardware details, memory (paging, DMA) configuration, driver status, interrupt configurations (usage), port information, network information, and printer information.

Total operating system failure procedure

The system administrator can configure how the system reacts in case of a total failure (dump).

Last known good configuration

In the event that the system administrator makes a mistake in changing configuration information, there is a feature known as the last known good menu; the last known good information from the last reboot is stored on the system until a user logs in. Every time Windows 2000 boots, the user is prompted to press the spacebar to enter the last known good menu. From this menu, as long as the system has not yet been logged on to since the errant configuration change, the administrator can restore the previous, or last known good, configuration.

Safe mode boot

Windows 2000 provides this functionality to help the system administrator solve the booting problems of the operating system. For a detailed description of Safe Mode Boot, see section 8.2.2.3, “Booting in Windows 2000 Safe Mode” on page 290.

Recovery console

This is a recovery and repair tool integrated in Windows setup. The system can be booted from an installation CD-ROM or from bootable floppies and the recovery console can be started by choosing the *Repair option* from the *Welcome* window. From the recovery console, the system administrator can run several commands, such as `FixMBR` (for fixing master boot record), `fdisk`, `format`, `copy`, `delete`, `dir`, `type`, and also enable and disable Windows 2000 Services.

Emergency Repair Process (ERP) and Emergency Repair Disk (ERD)

It is highly recommended that the system administrator create Emergency Repair Disk(s) (ERDs) that are bootable and that enable him or her to boot the system and repair system files, the startup environment in multiboot systems, and the partitions information.

Note

The emergency repair disk is system-dependent and cannot be used to recover other systems. It can only help recovering the system on which it was created.

8.2.7.8 Event logging and error logging

Windows 2000 defines an event as any significant occurrence in the system or in an application that users should be aware of and perhaps be notified about interactively. Windows 2000 provides a tool called the Event Viewer for reading and managing these events. For more details about Event Viewer, see section 8.2.4.2, “Windows 2000 Administrative Tools” on page 301.

8.2.8 Windows 2000 printing

From Windows 2000, we can print on printers attached to a local system, a remote system or to the network. We can also manage the printers attached to other systems, if we have the appropriate permissions. One of the advantages of the Windows 2000 printing subsystem is that you do not need to install a printer driver on a client computer since it is downloaded automatically from the print server the first time the printer is configured.

8.2.8.1 Windows 2000 printing terminology

Each operating system has its own terminology to describe internal processes. Microsoft uses the following terminology when it comes to printing in Windows 2000:

Printer (or Print Device)

The physical device that does the actual printing.

Logical Printer

This is the software interface on the print server. Each printer appears as a separate panel and is managed by using the Windows 2000 Control Panel option Printers.

Print Spooler

This is part of the operating system software that accepts a document sent for printing and stores it temporary in the system memory or on the disk until the printer is ready to print it. The term Spool (Spooler) is an acronym for "Simultaneous Print Operations On Line."

Printer Driver

This is a program that allows other programs to work with the actual physical printer without knowing its hardware and internal control software specifics.

Printer Pool

Two or more printers can be pointed to by one logical printer and behave as one printer. The document will be printed on the first available free printer.

Printer Queue

This is a list of printer jobs (documents sent for printing) awaiting execution by the *Print spooler*.

Printer Server

This is a computer that is designated and configured to manage one or more printers.

8.2.8.2 Windows 2000 printing process

The printing process is presented in Figure 112 on page 321.

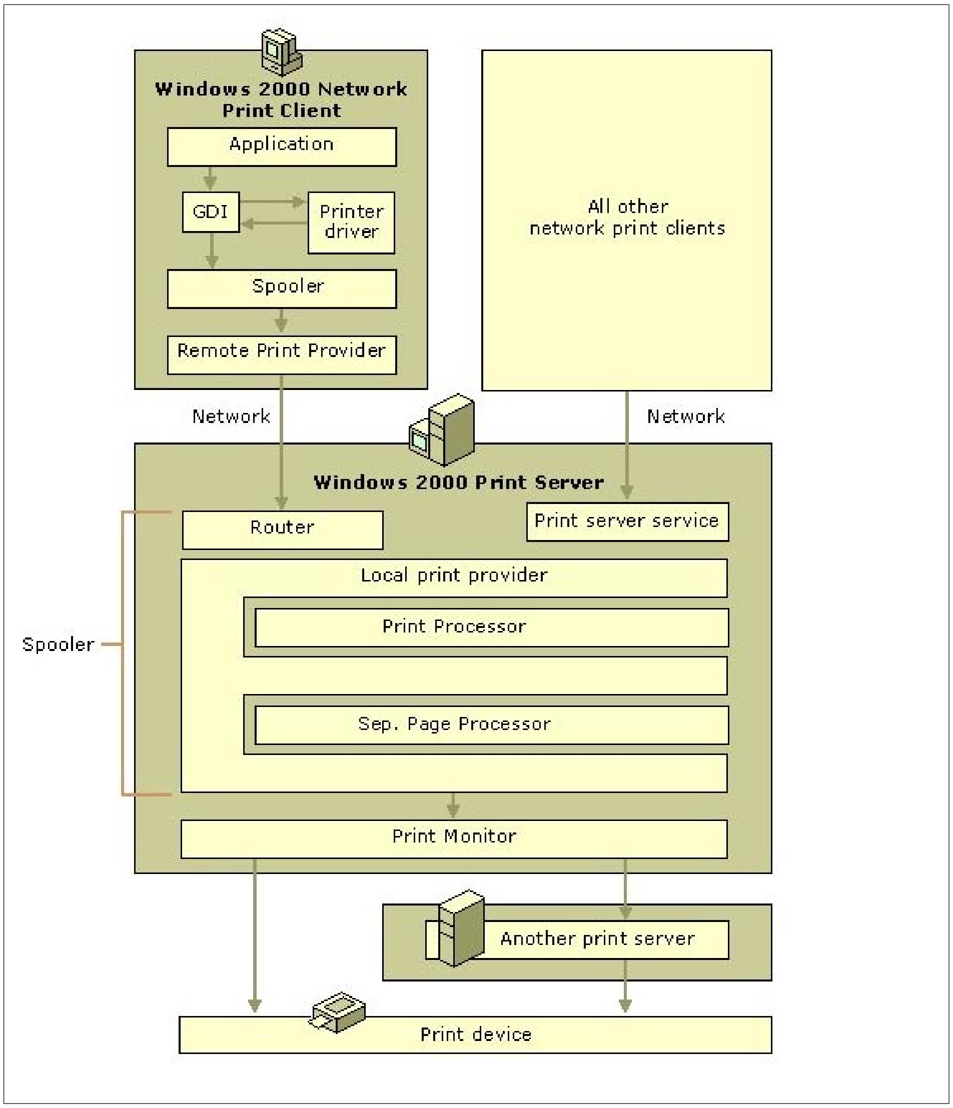


Figure 112. Windows 2000 printing process

When a user submits a document for printing the following procedure is performed:

1. In case that a user prints from Windows application, this application first calls *Graphics Device Interface* (GDI) which then calls the printer driver

associated with the printer. Depending on information contained in a submitted document, GDI and printer driver exchange information and translate the data to the form appropriate for the printer. The print job is then passed to client-side spooler. If the user is printing the document from other Non-Windows operating systems, other parts of the system software are responsible for formatting data (for example, *Virtual Printer* on AIX).

2. The client computer sends the print job to the print server. In Windows environments the *Remote Procedure Call* (RPC) functionality is used between client and print server to negotiate and send and receive print job.
   If the clients use Windows 2000 or Windows NT, the job is in Enhanced Metafile Format (EMF) data format. Most other clients use RAW data type.
3. The print router on the server sends the print job to the local print provider on the server, which actually spools the print job (temporarily writes it to the disk).
4. The local print provider polls the print processor. The print processor receives the print job, checks its data format, and converts it according to data type.

   If the target physical printer is defined on the client computer, the print server decides whether the print spooler should change the print jobs data type or assign it a different data type. The print job then passes to the local print provider, which writes it temporarily to the disk.
5. The control of the print job is then passed to the separator page processor, which adds a separator page (if so specified) as the first page of the document (job).
6. The print job is then send to print monitors. If we are using bi-directional printers, a monitor handles two-way communication between the sender and the printer and then passes the job to port monitor. If the printer is not bi-directional, the print jobs go directly to the port monitor.

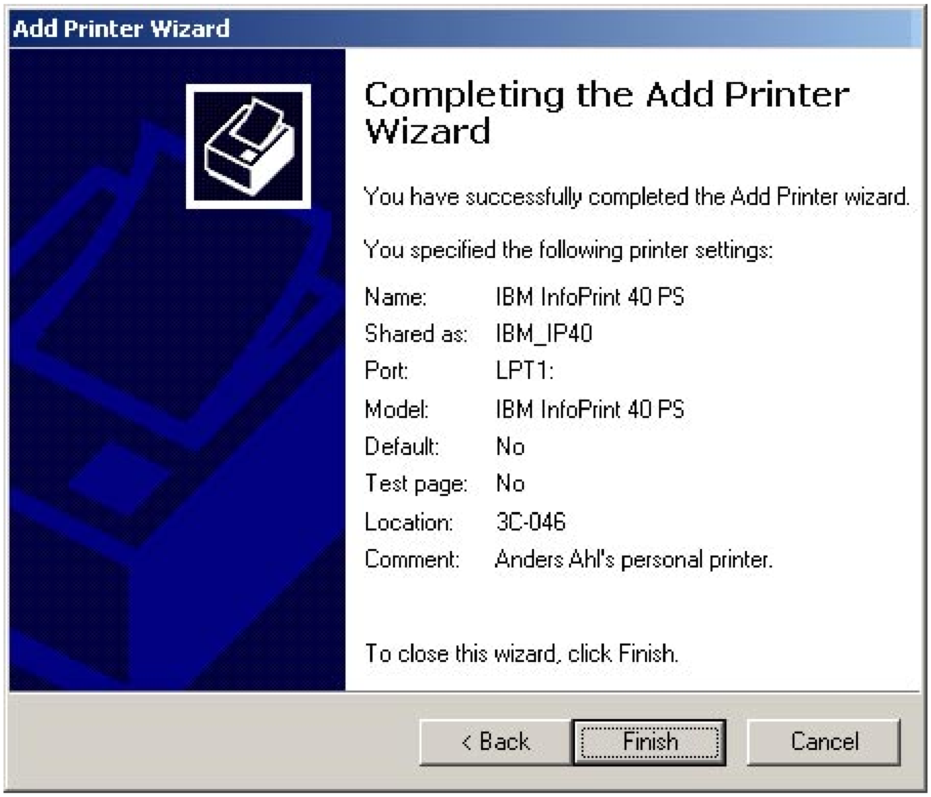
   Port monitor sends the job to the target, the actual physical printer.
7. The printer receives the job, converts it internally to the appropriate format (a bitmap for laser and ink-jet printers), and prints it on the print media (paper, transparencies, labels, and so on)

8.2.8.3 Windows 2000 print management

This section describes some common print manager tasks in Windows 2000 environment.

Adding a new printer

To add a new printer to a Windows 2000 machine, the system administrator clicks on the Add Printer icon, displayed in the printer panel of the Windows 2000 Control Panel. The *Add Printer* wizard is started, and, after answering several questions, the final confirmation screen similar to the one displayed in Figure 113 is shown.



Completing the Add Printer Wizard

You have successfully completed the Add Printer wizard.

You specified the following printer settings:

Name:	IBM InfoPrint 40 PS
Shared as:	IBM_IP40
Port:	LPT1:
Model:	IBM InfoPrint 40 PS
Default:	No
Test page:	No
Location:	3C-046
Comment:	Anders Ahl's personal printer.

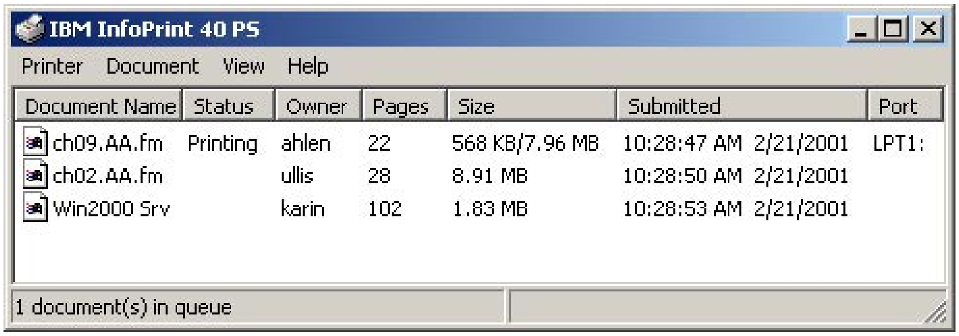
To close this wizard, click Finish.

Figure 113. Windows 2000 - Adding a printer

Managing print jobs

Print jobs can be managed before they are actually sent to a physical printer. The jobs can be stopped, canceled, or prioritized. The system administrator can start a print job management window by clicking on the particular printer in the printer panel of the Windows 2000 Control Panel (see “Printers” on page 299).

Depending on the number of jobs in the print queue, the window will look similar to one shown in Figure 114 on page 324.

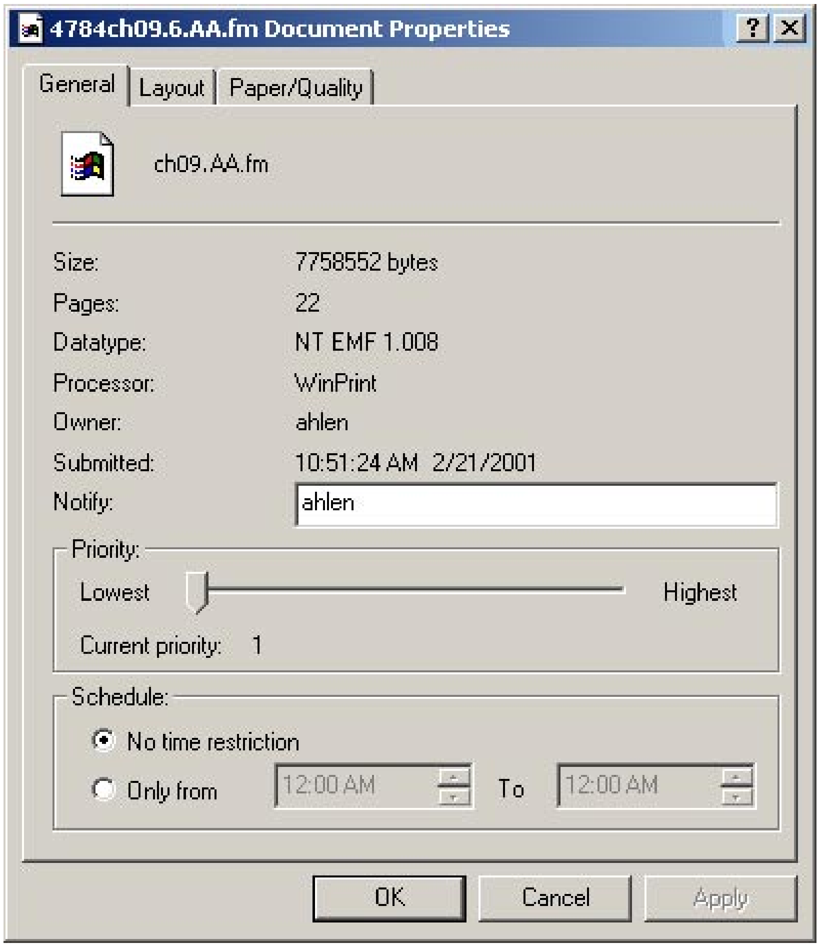


Document Name	Status	Owner	Pages	Size	Submitted	Port
ch09.AA.fm	Printing	ahlen	22	568 KB/7.96 MB	10:28:47 AM 2/21/2001	LPT1:
ch02.AA.fm		ullis	28	8.91 MB	10:28:50 AM 2/21/2001	
Win2000 Srv		karin	102	1.83 MB	10:28:53 AM 2/21/2001	

1 document(s) in queue

Figure 114. Windows 2000 - Print job management

By right clicking on the job and selecting properties, a new window is opened in which a printer administrator can change the properties of the print job (see Figure 115 on page 325).



4784ch09.6.AA.fm Document Properties

ch09.AA.fm

Size: 7758552 bytes
Pages: 22
Datatype: NT EMF 1.008
Processor: WinPrint
Owner: ahlen
Submitted: 10:51:24 AM 2/21/2001
Notify: ahlen

Priority:
Lowest ——— Highest
Current priority: 1

Schedule:
● No time restriction
○ Only from 12:00 AM To 12:00 AM

Figure 115. Windows 2000 - Print job properties

8.2.9 Windows 2000 Terminal services

Terminal services enables remote computers to access Windows 2000 programs running on the Windows 2000 Server. 16-bit and 32-bit Windows based clients are supported. Basically, it is terminal emulation; the user part of the program is sent to the client which, in return, sends keyboard and mouse input to the program that runs on the server.

One of the main advantages of Terminal services is that Windows 2000 applications can be used on a computer that is incapable of running Windows 2000 itself. Clients on other platforms, such as UNIX and Macintosh, can also connect to a Terminal server using additional third-party software such as Citrix Metaframe.

Terminal Services can be used in two different modes:

Application mode

This is the mode for using Windows applications on other computers. After installing the Remote Desktop Protocol (RDP) client on the nodes, the users can connect to the Terminal Server from anywhere with TCP/IP communication. It is not necessary for the client to even be in the same domain as the logon and authentication takes place in the RDP session itself.

Application mode requires a Terminal Services License server to be present on the network as separate licenses are required for RDP connections.

If Citrix Metaframe will be installed on the server to allow ICA connections, the server has to be in Application mode as well.

Remote administration mode

In this mode, Terminal services provides access for administrating the server from other systems on the network.

There is no need for a Terminal Services License server but only members of the Administrators group are allowed access to the server using RDP clients.

8.2.10 Windows 2000 support for national languages

Windows 2000 is localized and translated to the languages shown in Table 1 on page 58.

Localized versions of Windows 2000 consist of appropriate *Locales* (described below) and:

- Additional fonts
- Additional Input Method Editors (IMEs)
- Special tools
- Addition local device drivers (modems, printers)
- Legacy BIOS and DOS support
- Fully-localized user interface
- Localized documentation (Help files, readme files, release notes)

8.2.10.1 Windows 2000 Locales

In Windows 2000, a Locale is defined as a set of user preferences related to the user's language environment and cultural conventions. A Locale consist of keyboard layout, sorting order, format for representing dates, time, and currency symbols. In order to use one of the Locales, the appropriate Language Group must be installed on the system.

A Language Group is a set of keyboard layouts IMEs, fonts, and codepages necessary to support the given group of languages. The following Language Groups defined in Windows 2000:

- Arabic
- Armenian
- Baltic
- Central Europe
- Chinese Simplified
- Chinese Traditional
- Cyrillic
- Georgian
- Greek
- Hebrew
- Indic
- Japanese
- Korean
- Thai
- Turkish
- Vietnamese
- Western Europe (installed by default and cannot be removed)
- USA (installed by default and cannot be removed)

Windows 2000 differentiates between different types of Locales:

User Locale

This is a Locale specified for each and every user on the system. This however is not a language setting and has nothing to do with input languages, keyboard layouts, codepages, and user interface languages. It is used only to format dates, time, currency, numbers, and sorting order.

Default User Locale

This is the Locale applied by default when a new user (account) is created on the system.

Input Locales

This Locale is a pairing of an *input language* and an *input method*. It describes the language being entered and the way it is entered (keyboard, IME, speech-to-text converter).

System Locale

This Locale determines which ANSI, OEM, and MAC codepages and associated bitmap font files are used as a default for the system. These codepages and fonts enable non-Unicode applications to run as they would on a system localized to the language of system locale. The System Locales are supported for all supported Locales on all language versions. The setting of System Locale is valid for all users on the system.

8.2.10.2 Windows 2000 code pages

The following code pages are supported on Windows 2000:

- **SBCS** (single byte character set) codepages:
  - 1250 (Central Europe)
  - 1251 (Cyrillic)
  - 1252 (Latin I)
  - 1253 (Greek)
  - 1254 (Turkish)
  - 1255 (Hebrew)
  - 1256 (Arabic)
  - 1257 (Baltic)
  - 1258 (Vietnam Am)
  - 874 (Thai)
- **DBCS** (double byte character set) codepages:
  - 932 (Japanese Shift-JIS)
  - 936 (Simplified Chinese GBK)
  - 949 (Korean)
  - 950 (Traditional Chinese Big5)

8.2.10.3 Windows 2000 MUI

The Windows 2000 family of operating systems provides extensive support for international users as has been discussed in previous sections. This includes addressing multilingual issues such as keyboard layouts, sorting orders, date formats and fonts.

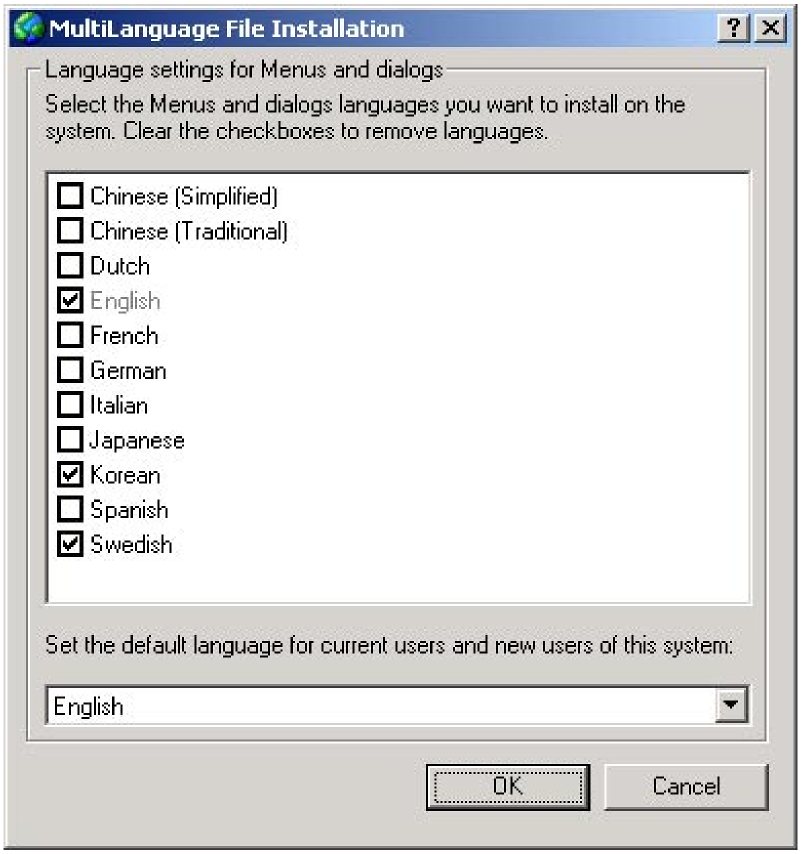
The Windows 2000 Multi-language version, available only through Volume Licensing programs such as the Microsoft Open License Program (MOLP / Open), Select, and Enterprise agreement, builds on top of this support by adding the capability of switching the User Interface (menus, dialogs and help files) from one language to another.

Using this feature instead of completely localized versions of Windows 2000 makes administration of multilingual computing environments a much easier task.

Each machine running Windows 2000 Professional must be prepared with the necessary language packs which can either be installed manually by the administrator or deployed in parallel with the Windows 2000 operating system.

In this example, we will manually add two languages to a Windows 2000 Professional machine effectively making it usable for all three nationalities of this redbook team.

The MUI files (language packs) come on two CDs with the Korean and Swedish packages both on CD2. Running `MUISETUP.EXE` from the root of CD2 presents the window in Figure 116 where we have selected Korean and Swedish. Since the installed version of Windows 2000 is English, this language is already checked and greyed out.



MultiLanguage File Installation

Language settings for Menus and dialogs

Select the Menus and dialogs languages you want to install on the system. Clear the checkboxes to remove languages.

☐ Chinese (Simplified)
☐ Chinese (Traditional)
☐ Dutch
☑ English
☐ French
☐ German
☐ Italian
☐ Japanese
☑ Korean
☐ Spanish
☑ Swedish

Set the default language for current users and new users of this system:

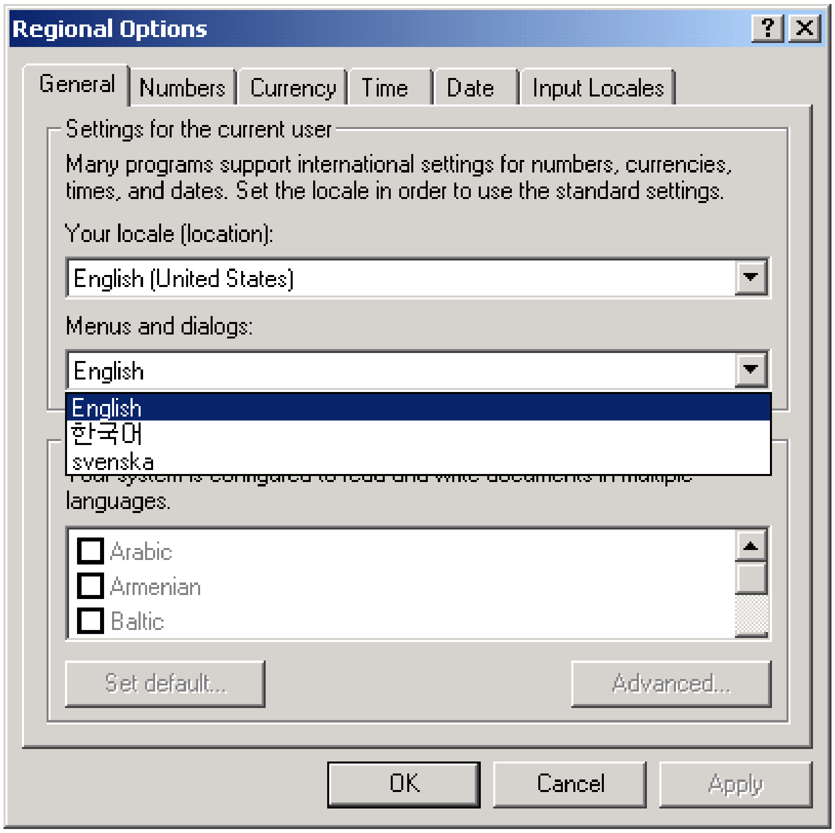
English

Figure 116. Multi-Language File Installation

If we change the default language for users (as seen at the bottom of Figure 116 on page 329), the login dialog will change language as well. We will leave this at English as to accommodate all possible users attempting to log on to this machine.

The MUI files are installed in the %SystemRoot%\MUI directory so if we want to add or remove languages or change default language setting at a later point in time we can run the `MUISETUP.EXE` program again.

After a mandatory reboot, we log on with a user account, open up the Control Panel and select Regional Options. Notice that everything is still in English since this is the default language and we have not changed any language parameters for our user. Selecting the Menus and dialogs selection box as seen in Figure 117, the three different languages we can choose from are shown. Note that this selection box is not shown at all if you do not have additional languages installed on your system.



Many programs support international settings for numbers, currencies, times, and dates. Set the locale in order to use the standard settings.

Your locale (location): English (United States)

Menus and dialogs: English

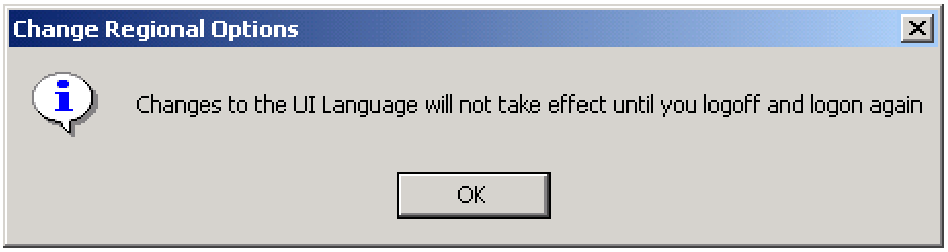
Your system is configured to read and write documents in multiple languages.

□ Arabic
□ Armenian
□ Baltic

Figure 117. Regional Options - Menus and dialogs

We select “svenska” which is “Swedish” in English and we also change the locale to Swedish which will give us the correct time and date representation. Before we change the Input Locales we logoff and logon to show the changes in dialogs. In a real-world situation you would change the input locale at the same time to avoid starting the Control Panel again after logging on.

When we select OK or Apply, the window in Figure 118 appears, prompting us to logoff and on for the changes to take effect.



Changes to the UI Language will not take effect until you logoff and logon again

Figure 118. Change Regional Options

After doing this and subsequently opening the Control Panel, we get the result from Figure 119 on page 332. Notice how most of the applets are translated into Swedish but there are still six applets with English names. This is perfectly alright as not all parts of the operating system will be localized. Installing a Swedish version of Windows 2000 would give more applets Swedish names but there would still be entries in English, especially if you have installed third-party services that do not support localization.

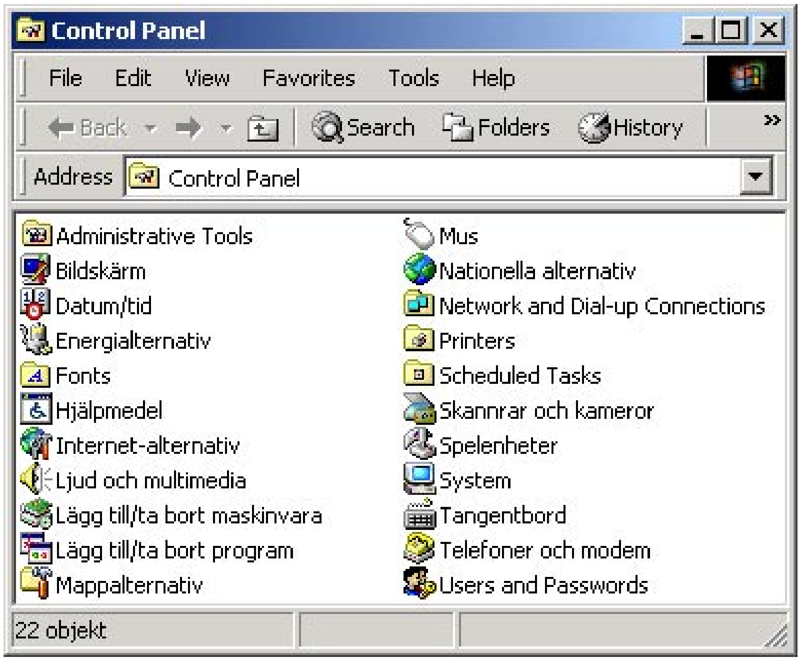
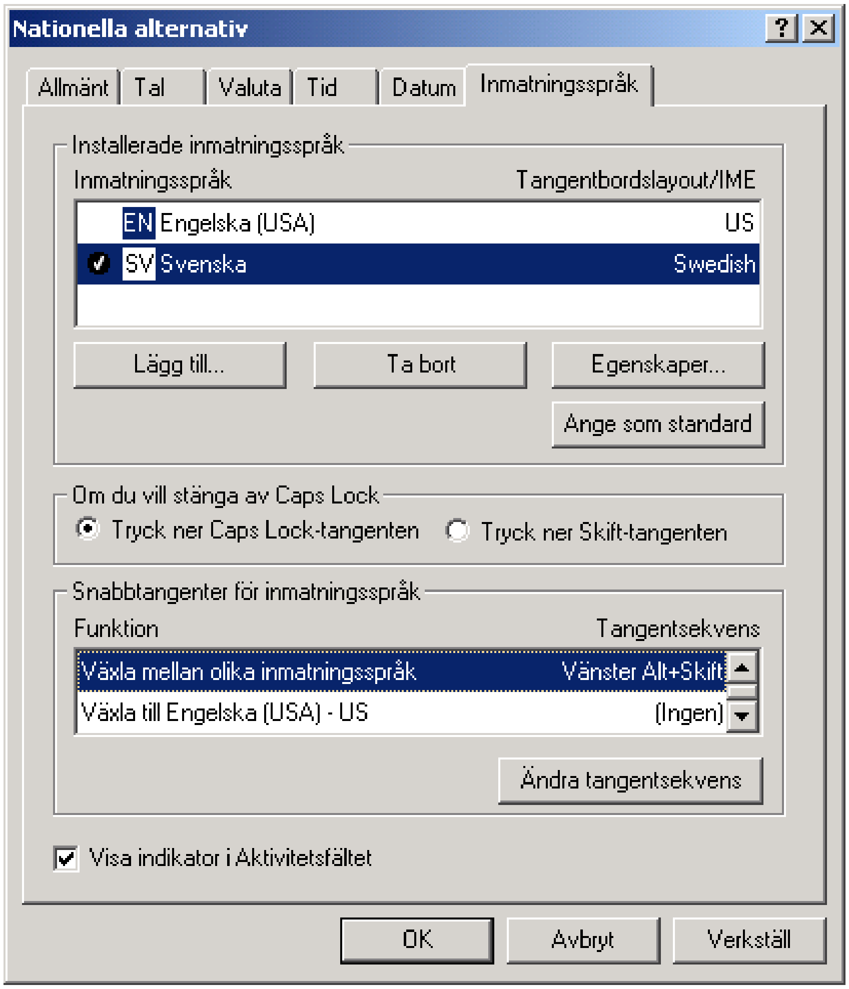


Figure 119. Control Panel with Swedish dialogs

By starting the Regional Options again, this time called “Nationella alternativ“ in Swedish (you can probably find it by looking at the icons), we select the right-most tab which is the Input Locales (inmatningssprak) and select Svenska as the default keyboard as seen in Figure 120 on page 333.



Installerade inmatningsspråk

Inmatningsspråk	Tangentbordslayout/IME
EN Engelska (USA)	US
✓ SV Svenska	Swedish

Om du vill stänga av Caps Lock

● Tryck ner Caps Lock-tangenten ○ Tryck ner Skift-tangenten

Snabbtangenter för inmatningsspråk

Funktion	Tangentsekvens
Växla mellan olika inmatningsspråk	Vänster Alt+Skift
Växla till Engelska (USA) - US	(Ingen)

☑ Visa indikator i Aktivitetsfältet

Figure 120. Input Locales

When we press OK, the user is all set for a Swedish workstation. Logging on with a new user, without a localized profile will get the default language which is English.

Going through the same procedure again, but changing all settings to Korean yields the Control Panel seen in Figure 121 on page 334.

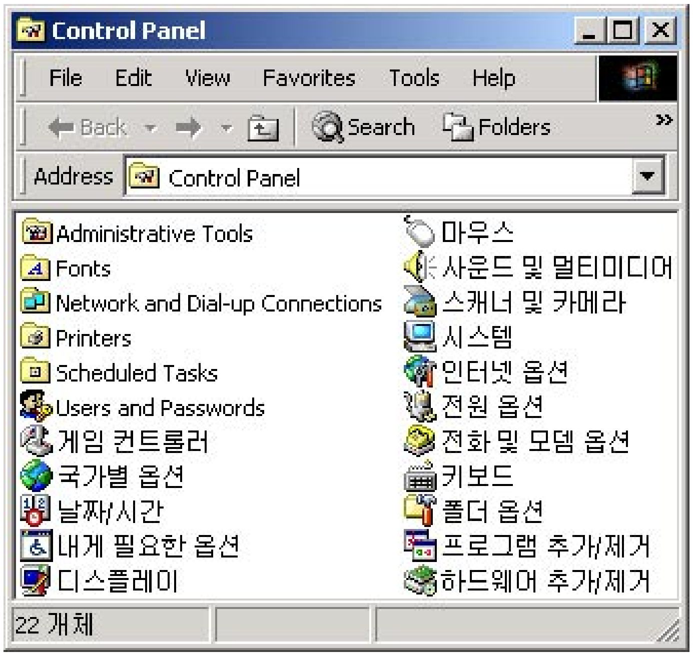


Figure 121. Control Panel with Korean dialogs

As long as we keep the default to English, the login prompt will still be in English, which is highly recommended, especially when you involve DBCS languages.

Chapter 9. Performance monitoring

Optimizing performance and finding performance bottlenecks is one of the major concerns of a system administrator. In particular, the following areas are very important for keeping a system well-tuned and performing properly:

- **Load Monitoring** - Resource load must be monitored so performance problems can be detected as they occur or (preferably) predicted before they do.
- **Analysis and Control** - Once a performance problem is encountered, the proper tools must be selected and applied so that the nature of the problem can be understood and corrective action taken.
- **Capacity Planning** - Long-term capacity requirements must be analyzed so that sufficient resources can be acquired well before they are required.

AIX 5L and Windows 2000 offer a set of tools and utilities that help a system administrator face and resolve these kinds of issues. In this chapter, we are going to discuss and present the solutions provided by both operating systems.

9.1 AIX performance monitoring

Due to its UNIX heritage, AIX provides several performance monitoring and tuning tools. These tools give you information on the performance of various aspects of the system and on the parameters that affect performance. For reader convenience, these tools have been grouped into five categories: CPU, I/O, Memory, Network, and Others.

9.1.1 Monitoring CPU

The following is a list of standard UNIX tools that allow CPU performance analysis.

ps

The `ps` command displays statistics and status information about the processes in the system, such as Process ID, I/O activity, and CPU utilization.

vmstat

The `vmstat` command reports statistics about kernel threads, virtual memory, disks, traps, and CPU activity. The activity is a percentage breakdown of user mode, system mode, idle, and waiting for disk I/O.

sar

The `sar` command writes the contents of selected cumulative activity counters in the operating system to standard output. The accounting system, based on the values in the number and interval parameters, writes information the specified number of times spaced at the specified intervals in seconds. The `sar` command reports either system wide (global among all processors) statistics, which are calculated as averages for values expressed as percentages and as sums, or it reports statistics for each individual processor.

> **Note**
>
> The `sar` command only reports on local activities.

The following values are reported by the `sar` command:

- The number of kernel processes terminating per second
- The number of times kernel processes could not be created because of enforcement of the process threshold limit
- The number of kernel processes assigned to tasks per second
- The number of IPC message primitives and semaphore primitives
- Queue, paging, CPU, and tty statistics

9.1.2 Monitoring I/O

The following is a list of UNIX tools that allow I/O performance monitoring.

filemon

This tool monitors the performance of the file system and reports the I/O activity on behalf of logical files, virtual memory segments, Logical Volumes, and Physical Volumes. The `filemon` command monitors a trace of file system and I/O system events and reports on the file and I/O access performance during that period. To provide a more complete understanding of file system performance for an application, the `filemon` command monitors file and I/O activity at four levels:

- **Logical File System** - The `filemon` command monitors logical I/O operations on logical files. The monitored operations include all read, write, open, and lseek system calls, which may or may not result in actual physical I/O, depending on whether or not the files are already buffered in memory. I/O statistics are kept on a per-file basis.

- **Virtual memory system** - The `filemon` command monitors physical I/O operations (that is, paging) between segments and their images on disk. I/O statistics are kept on a per-segment basis.
- **Logical Volumes** - The `filemon` command monitors I/O operations on Logical Volumes. I/O statistics are kept on a per-logical-volume basis.
- **Physical Volumes** - The `filemon` command monitors I/O operations on Physical Volumes. At this level, physical resource utilizations are obtained. I/O statistics and kept on a per-Physical Volume basis.

Any combination of the four levels can be monitored. Some of the fields report a single value. Others report statistics that characterize a distribution of many values. For example, response time statistics are kept for all read or write requests that were monitored. The average, minimum, and maximum response times are reported, as well as the standard deviation of the response times.

iostat

The `iostat` command is used for monitoring system input/output device loading by observing the time the physical disks are active in relation to their average transfer rates. The `iostat` command generates reports that can be used to change system configuration to better balance the input/output load between physical disks. The `iostat` command is useful in determining whether a Physical Volume is becoming a performance bottleneck and if there is potential to improve the situation. Since the CPU utilization statistics are also available with the iostat report, the percentage of time the CPU is in I/O wait can be determined at the same time. For multiprocessor systems, the CPU values are global averages among all processors. Also, the I/O wait state is defined system wide and not per processor. The report shows several parameters:

- The total number of characters read/written by the system for all ttys.
- The percentage of CPU utilization that occurred while executing at the user/system level (application/kernel).
- The percentage of time that the CPU or CPUs were idle and the system did not have an outstanding disk I/O request.
- The percentage of time that the CPU or CPUs were idle during which the system had an outstanding disk I/O request. This value may be slightly inflated if several processors are idling at the same time (an unusual occurrence).
- The percentage of time the physical disk was active (bandwidth utilization for the drive).

- The amount of data transferred (read or written) to the drive in KB per second.
- Indicates the number of transfers per second that were issued to the physical disk. A transfer is an I/O request to the physical disk
- The total number of KB read/written.
- Statistics for CD-ROM devices.
- In AIX 5L, using -i flag, the output shows the disk adapter activity and a per-disk basis set of statistics.

This information is updated at regular intervals by the kernel (typically sixty times per second).

9.1.3 Monitoring memory

The following is a list of UNIX tools that allow memory monitoring.

svmon

The `svmon` command displays information about the current state of memory. The displayed information does not constitute a true snapshot of the memory because the `svmon` command runs at user level with interrupts enabled. The `svmon` command creates four types of reports: Global, process, segment, and detailed segment. The reports are:

- Global statistics describing the use of real memory
- Statistics on the subset of real memory in use
- Statistics on the subset of real memory containing pinned pages
- Statistics describing the use of paging space

In AIX 5L, `svmon` command has been enhanced to display information about different superclasses and subclasses introduced with Workload manager.

vmstat

The `vmstat` command reports statistics about kernel threads, virtual memory, disks, traps, and CPU activity. Reports generated by the `vmstat` command can be used to balance system load activity. These system wide statistics (among all processors) are calculated as averages for values expressed as percentages, or, otherwise, as sums. The `vmstat` command accesses statistics maintained by the kernel, such as kernel threads, paging, interrupt activity, and statistics maintained by the device drivers, such as disk input/output. For example the average disk transfer rate is determined by using the active time and number of transfers information. The percent active time is computed from the amount of time the drive is busy during the report. Some statistics are:

- Kernel thread state changes per second over the sampling interval
- Number of kernel threads placed in the run or wait queue (awaiting resource, awaiting input/output)
- Active virtual pages
- Size of the free list
- Pager input/output list
- Pages paged in/out from paging space or freed
- Pages scanned by page-replacement algorithm
- Clock cycles by page-replacement algorithm
- Device interrupts and system calls
- Kernel thread context switches
- User/System/CPU_idle time
- CPU cycles to determine that the current process is waiting and there is a pending disk input/output

The following screen output shows the result of `vmstat` command with two options, interval (2 seconds) and count (5 times).

```
# vmstat 2 5
kthr     memory             page              faults        cpu
----- ----------- ------------------------ ------------ -----------
 r  b   avm   fre  re  pi  po  fr   sr  cy  in   sy  cs us sy id wa
 0  0 32975  1030   0   0   0   8   21   0 125  771  82  3  1 95  2
 0  1 32975  1028   0   0   0   0    0   0 117 3075  80  2  2 92  4
 0  1 32975  1028   0   0   0   0    0   0 116   42  31  0  0 99  0
 0  1 32975  1028   0   0   0   0    0   0 116   30  31  0  0 99  0
 0  1 32975  1028   0   0   0   0    0   0 115   24  30  0  0 99  0
```

The vmstat utility has two new flags in AIX 5L; these new flags add new controls and improve monitoring.

The -I flag outputs a report with the new columns *fi* and *fo*; these columns indicate the number of file pages in (*fi*) and out (*fo*). In this report, the *re* and *cy* columns are not displayed. A new *p* column displays the number of threads waiting for a physical I/O operation as shown in following screen.

```
 vmstat -I 2 5
  kthr      memory             page              faults        cpu
-------- ----------- ------------------------ ------------ -----------
 r  b  p   avm   fre  fi  fo  pi  po  fr   sr   in   sy  cs us sy id wa
 0  0  0 33408  1017   0  12   0   0   8   21 125  762  81  2  1 93  4
 0  2  0 33408  1015   0   0   0   0   0    0 117 3802  37  1  3  0 96
 0  2  0 33408  1015   0   0   0   0   0    0 120   43  35  0  0  0 99
 0  2  0 33408  1015   0   0   0   0   0    0 117   28  30  0  0  0 99
 0  2  0 33408  1015   0   0   0   0   0    0 116   26  29  0  0  0 99
```

The -t flag shows a timestamp at the end of each line as below screen.

```
# vmstat -t 2 3
kthr     memory             page              faults        cpu        time
----- ----------- ------------------------ ------------ ----------- --------
 r  b   avm   fre  re  pi  po  fr   sr  cy  in   sy  cs us sy id wa hr mi se
 0  0 32996  2217   0   0   0   8   21   0 125  759  81  2  1 92  4 17:53:42
 0  1 32996  2214   0   0   0   0    0   0 117 3395  32  2  2 96  0 17:53:44
 0  1 32996  2214   0   0   0   0    0   0 118   42  36  0  0 99  0 17:53:46
```

ps

The ps command can be used to monitor the memory usage of an individual process. The ps v [pid] command provides reports on memory-related statistics for an individual process, such as page faults, the size of working segments that have been touched, the size of working segments and code segments in memory, the size of text segments, the size of the resident set, and the percentage of real memory used by this process.

rmss

rmss is an acronym for Reduced-Memory System Simulator. rmss provides you with a means of simulating RISC System/6000s with different sizes of real memories that are smaller than your actual machine without having to extract and replace memory boards. Moreover, rmss provides a facility to run an application over a range of memory sizes, displaying, for each memory size, performance statistics such as the response time of the application and the amount of paging. In short, rmss helps answer how many megabytes of real memory are needed to run AIX on a given application or how many users can run this application simultaneously in a machine with X megabytes of real memory. It is important to keep in mind that the memory size simulated by rmss is the total size of the machine’s real memory, including the memory used by AIX and any other programs that may be running. It is not the amount of memory used specifically by the application itself. Because of the performance degradation it can cause, rmss can be used only by root or a member of the system group.

9.1.4 Monitoring the network

The following is a list of UNIX tools that allow network performance monitoring.

netstat

The `netstat` command displays information regarding traffic on the configured network interfaces, such as:

- The address of any protocol control blocks associated with the sockets and the state of all sockets
- The number of packets received, transmitted, and dropped in the communications subsystem
- Cumulative statistics for Error Collision Packets transferred
- Routes and their statuses

nfsstat

The `nfsstat` command displays statistics on Network File System (NFS) and Remote Procedure Call (RPC) server and client activity, such as:

- **NFS server information** - The NFS server displays the number of NFS calls received (calls) and rejected (badcalls), as well as the counts and percentages for the various kinds of calls made.
- **NFS client information** - The NFS client displays the number of calls sent and rejected, as well as the number of times a client handle was received (nclget), the number of times a call had to sleep while awaiting a handle (nclsleep), and a count of the various kinds of calls and their respective percentages.
- **RPC statistics** - The `nfsstat` command displays statistical information pertaining to the ability of a client or server to receive calls, including:
  - The total number of RPC calls received or rejected
  - The number of times no RPC packet was available when trying to receive
  - The number of packets that were too short or had a malformed header
  - The total number of RPC calls sent or rejected by a server
  - The number of times a call had to be transmitted again
  - The number of times a reply did not match the call
  - The number of times a call timed out
  - The number of times a call had to wait on a busy client handle
  - The number of times authentication information had to be refreshed

9.1.5 Other tools

The following is a list of additional useful performance monitoring tools.

trace

The AIX trace facility is useful for observing a running device driver and system. The trace facility captures a sequential flow of time-stamped system events, providing a fine level of detail on system activity. Events are shown in time sequence and in the context of other events. The trace facility is useful in expanding the trace event information to understand who, when, how, and even why the event happened. The operating system is shipped with permanent trace event points. These events provide general visibility to system execution. You can extend the visibility into applications by inserting additional events and providing formatting rules with low overhead. Because of this, the facility is useful as a performance-analysis tool and as a problem-determination tool.

The trace facility is more flexible than traditional system monitor services that access and present statistics maintained by the system. With traditional monitor services, data reduction (conversion of system events to statistics) is largely coupled to the system instrumentation. For example, the system can maintain the minimum, maximum, and average elapsed time observed for runs of a task and permit this information to be extracted. The trace facility does not strongly couple data reduction to instrumentation, but provides a stream of system events. It is not required to presuppose what statistics are needed. The statistics or data reduction are, to a large degree, separated from the instrumentation. You can choose to develop the minimum, maximum, and average time for task A from the flow of events, but it is also possible to extract the average time for task A when called by process B, extract the average time for task A when conditions XYZ are met, or even decide that some other task, recognized by a stream of events, is more meaningful to summarize. This flexibility is important for diagnosing performance or functional problems. For example, netpmon uses the trace to report on network activity, including CPU consumption, data rates, and response time. tprof uses the trace to report the CPU consumption of kernel services, library subroutines, application-program modules, and individual lines of source code in the application program.

PDT

The Performance Diagnostic Tool (PDT) collects configuration and performance information and attempts to identify potential problems that may arise currently or in the future. In assessing the configuration and the historical record of performance measurements, PDT attempts to identify:

- Imbalanced use of resources or asymmetrical aspects of configuration or device utilization. In general, if there are several resources of the same type, the balanced use of those resources produces better performance:
  - Comparable numbers of Physical Volumes (disks) on each disk adapter
  - Paging space distributed across multiple Physical Volumes
  - Roughly equal measured load on different Physical Volumes
- Trends in usage levels that will lead to saturation. Resources have limits to their use. Trends that would attempt to exceed those limits should be detected and reported. For example, a disk drive cannot be utilized more than 100 percent of the time and file and file system sizes cannot exceed the allocated space.
- New consumers of resource-expensive processes that have not been observed before. Trends can indicate a change in the nature of the workload as well as increases in the amount of resources used, such as the number of users logged on, the total number of processes, and the CPU-idle percentage.
- Inappropriate system parameter value settings that may cause problems.
- Errors in hardware or software may lead to performance problems; so, it checks the hardware and software error logs and reports bad VMM pages.

Other commands are available for tuning your system resources. The system administrator can, for example, change the values of VMM memory load control parameters, the CPU-time-slice duration, and the paging-space-low retry interval with the `schedtune` command. Also, the system administrator can change the Virtual Memory Manager page-replacement algorithm parameters with the `vmtune` command, optimize executable files for a specific workload with the `fdpr` command, change the values of network options with the `no` command, change the priority of running processes with the `renice` command, display information about kernel lock contention with the `lockstat` command, count the system calls, and so on.

FDPR

Feedback Directed Program Restructuring (FDPR) is an optimization tool that improves program code locality. The tool receives input files in XCOFF format, instruments them, executes them for profiling information and then reorders them in order to get a better cache ratio.

A new improvement introduced with AIX 5L is the Code Duplication optimization. Code Duplication optimization eliminates the need to invoke the store and restore functions of small, but frequently used functions in the Link Register, which were not suitable for optimization, by creating a new copy of the called function and redirecting the calling instructions to its duplicated copy.

More information about this tool can be found in the AIX documentation, such as the article Restructuring Executables with fdpr in the book *AIX 5L Version 5.1 Performance Management Guide*.

pprof

This is a lightweight, trace-based tool that collects a system's process and thread information. Reports are generated in several formats, including a family view. The family view displays all parent-child relationships for all processes and threads. This tool is especially helpful in pinpointing system degradation when caused by multiple processes.

tprof

The `tprof` command reports CPU usage for individual programs and the system as a whole. This command is a useful tool for anyone with a C or C++ or FORTRAN program that might be CPU-bound and who wants to know which sections of this program are most heavily using the CPU. The `tprof` command also reports the fraction of time the CPU is idle. These reports can be useful in determining CPU usage in a global sense.The `tprof` command specifies the user program to be profiled, executes the user program, and then produces a set of files containing reports. tprof operates in two modes. The first is called the online mode, which has tprof execute the system `trace` command and a specific program, if specified, that is to be profiled. After the trace has completed, tprof processes the trace data and produces report files. In the offline mode, the trace data has already been gathered, and tprof simply reads from this file and, as before, processes the trace data and produces report files. If multiple trace files exist from multiple CPUs, tprof can also provide you with CPU-specific data.

Topas

A new performance monitoring program based on libspmi.a shared library is introduced with AIX 4.3.3. The program, topas, is the curses-based application that displays top processes and disk and memory usage. As its name implies, the topas program is a clone of the freeware program *top*. In AIX 5L, it has several new enhancements, including Workload Manager support, an improved set of CPU usage panels, several new column sort options, NFS statistics, lock statistics, and per disk or adapter breakdown of network and disk usage. The following screen provides a sample topas main output. It is beyond the scope of this section to demonstrate all the features. It is recommended that the topas tool is given a complete exploration through hands-on use.

```
Network  KBPS   I-Pack  O-Pack   KB-In  KB-Out  Waitqueue   1.0
Topas Monitor for host:    rs9916a              EVENTS/QUEUES    FILE/TTY
Thu Feb 22 17:33:11 2001   Interval:  2         Cswitch      26  Readch        0
                                                Syscall      28  Writech      59
Kernel    0.0   |                            |  Reads         0  Rawin         0
User    100.0   |############################|  Writes        0  Ttyout        0
Wait      0.0   |                            |  Forks         0  Igets         0
Idle      0.0   |                            |  Execs         0  Namei         0
                                                Runqueue    1.0  Dirblk        0
Network  KBPS   I-Pack  O-Pack   KB-In  KB-Out  Waitqueue   1.0
tr0       0.0      0.9     0.4     0.0     0.0
lo0       0.0      0.0     0.0     0.0     0.0  PAGING           MEMORY
                                                Faults        0  Real,MB     511
Disk    Busy%     KBPS     TPS KB-Read KB-Writ  Steals        0  % Comp     24.0
hdisk0    0.0      0.0     0.0     0.0     0.0  PgspIn        0  % Noncomp   9.0
hdisk1    0.0      0.0     0.0     0.0     0.0  PgspOut       0  % Client    0.0
                                                PageIn        0
WLM-Class (Active)     CPU%    Mem%  Disk-I/O%  PageOut       0  PAGING SPACE
batch                    99       0         0   Sios          0  Size,MB       0
System                    0      20         0                    % Used      0.6
                                                NFS (calls/sec)  % Free     99.3
Name         PID CPU% PgSp Class                ServerV2       0
cpuload    14464 99.5  0.0 batch                ClientV2       0   Press:
topas      16526  0.5  0.6 System               ServerV3       0   "h" for help
gil         1032  0.0  0.0 System               ClientV3       0   "q" to quit
```

truss

AIX 5L now supports the `truss` command, which allows you to trace system calls executed by a process as well as record the received signals and the occurrence of machine faults.

The application to trace is either specified on the command line of the `truss` command or `truss` can be attached to one or more already running processes by using the -p flag with a list of process IDs.

alstat

alstat is a new tool which reports alignment exception statistics. This tool can be used to detect performance degradations caused by misalignment data or code (POWER only).

9.1.6 Performance Toolbox for AIX

Performance Toolbox for AIX is a Licensed Program Product (not included in AIX) that offers a wide range of performance tools within a versatile framework for both stand-alone and networked systems. It includes a Motif-based toolbox, a 3D color graphic performance monitor application, and a data consumer applications program interface.

9.1.6.1 Features and benefits

PTX provides following useful functionalities and advantages.

Distributed Monitoring

With hundreds of available metrics, PTX provides an easy way of monitoring your local and remote systems as well as the performance of the network, which is paramount in SP and client/server environments.

Logfiles and graphical analysis for system performance

In distributed environments, problems can arise between systems working together or locally within machines. Furthermore, these same problems may appear as short spikes of activity or prolonged over a longer period of time. Therefore, it is very important to be able to get the "big picture.” PTX provides the user with the power to construct customizable graphs in 2 and 3 dimensions for either live data or recorded sessions. Logfiles can also be converted to a tabulated text format readable by spreadsheet applications. With a fully configurable graphical interface, PTX gives the user the ability to concurrently visualizes live performance characteristics and pinpoint either local, distributed or network bottlenecks, either during particularly intense periods, or over an extended period of time.

Analysis and control of system performance

By providing tools that can be used to analyze performance data and control system resources, PTX assists the system administrator in keeping track of available tools and in applying them in appropriate ways. This is done through a customizable menu interface. Tools can be added to menus, either with fixed parameters to match specific situations or in a dialog window.

Automated monitoring

Often situations arise which need immediate attention. PTX allows the user to define conditions and appropriate responses, which include alerting a specific administrator to initiating a corrective action without any human response.

9.1.6.2 PTX components

The Performance Toolbox for AIX is divided into two components:

- Performance Toolbox Agent
- Performance Toolbox Manager

The two components constitute the server (Agent) and the client (Manager) sides of a set of performance management tools, which allow performance monitoring and analysis in a networked environment.

The four main programs of the Manager component are all X Window System-based programs developed with the OSF/Motif Toolkit. One program, xmperf, is at the same time a graphical monitor, a tool for analyzing system load, and an umbrella for other tools, performance related or not. Another monitoring program is 3dmon, which allows the monitoring of a large number of statistics in a single window. The program exmon is designed to work with the filtd daemon. It monitors alarms generated by filtd. The fourth main program is the azizo program to analyze recordings of performance data.

The Performance Toolbox Manager has three packages:

- **perfmgr.local** This package contains the commands and utilities that allow monitoring of only the local system.
- **perfmgr.network** This package contains the commands and utilities that allow monitoring of remote systems as well as the local system.
- **perfmgr.common** This package contains the commands and utilities that are common between the network support and the local support.

The main program in the Agent component is the daemon xmservd, which acts as a networked supplier of performance statistics and, optionally, as a supplier of performance statistics to Simple Network Management Protocol (SNMP) managers.

The Performance Toolbox Agent has one package:

- **perfagent.server** This package contains the performance agent component required by Performance Toolbox as well as some local AIX analysis and control tools.

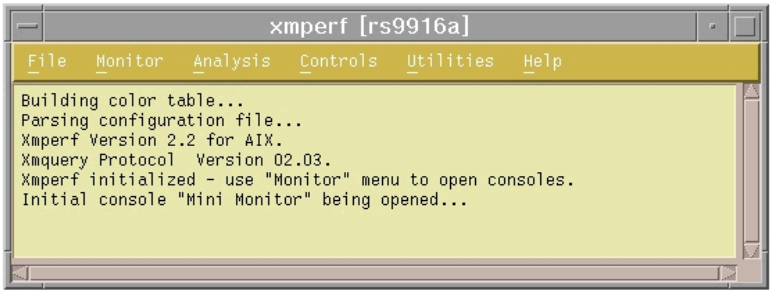
Finally there is a tool package, called perfagent.tools, which collects those pieces that are required to be built with the AIX kernel. The perfagent.tools contains the following utilities (for more details on these utilities, refer to previous paragraphs in this chapter):

- **bf** traces the memory access patterns of running programs (32-bit applications only).
- **fdpr** restructures programs based on observed execution.
- **filemon** traces detailed disk I/O activity.
- **fileplace** displays file block placement within a logical or physical volume.
- **lockstat** reports of contention for locks
- **netpmon** traces detailed network I/O activity

- **rmss** simulates smaller real memory to assess the memory requirements of programs
- **svmon** reports the current use of real and virtual memory
- **tprof** reports the CPU use of individual programs, subroutines, and system components

9.1.6.3 Monitoring features

The client/server environment allows any program, whether it is part of Performance Toolbox for AIX or custom-developed applications, to monitor the local host as well as multiple remote hosts. This ability is fully explored in the Manager component program, xmperf, whose *monitors* are graphical windows, referred to as consoles, that can be customized on the fly or kept as preconfigured consoles that can be invoked with a few mouse clicks. Consoles can be generic; so, the actual resource to monitor, whether it is a remote host, a disk drive, or a LAN interface, is chosen when the console is opened. Consoles can be told to do a recording to disk files of the data they monitor, and such recordings can be played back with xmperf and analyzed with the azizo program. Figure 122 on page 348 shows the main window of the xmperf tool.



```
Building color table...
Parsing configuration file...
Xmperf Version 2.2 for AIX.
Xmquery Protocol  Version 02.03.
Xmperf initialized - use "Monitor" menu to open consoles.
Initial console "Mini Monitor" being opened...
```

Figure 122. xmperf main window

One of the things that makes xmperf unique is that it is not hardcoded to monitor a fixed set of resources. It is dynamic in the sense that a system administrator can customize it to focus on exactly the resources that are critical for each host that must be monitored.

From the main menu bar of this window, it is possible to:

- Monitor some selected resources from the local or remote systems and display them in graphical windows
- Run analysis tools, such as tprofs, svmon, vmstat, rmss, PDT, filemon, iostat, or trace, just to list some of them

- From the Controls menu, interact with the processes running on the system and have easy control over them. In particular, it is possible to:
  - View all the process sorted in different ways
  - Execute some operations on them, such as generating an svmon report, killing one or more processes, or changing the priority of one or more process
  - Tune networking parameters
  - Verify and tune executables
- Access some utilities, such as 3dmon and azizo, have easy access to the system tables, and have control over the `xmperfd` daemon.

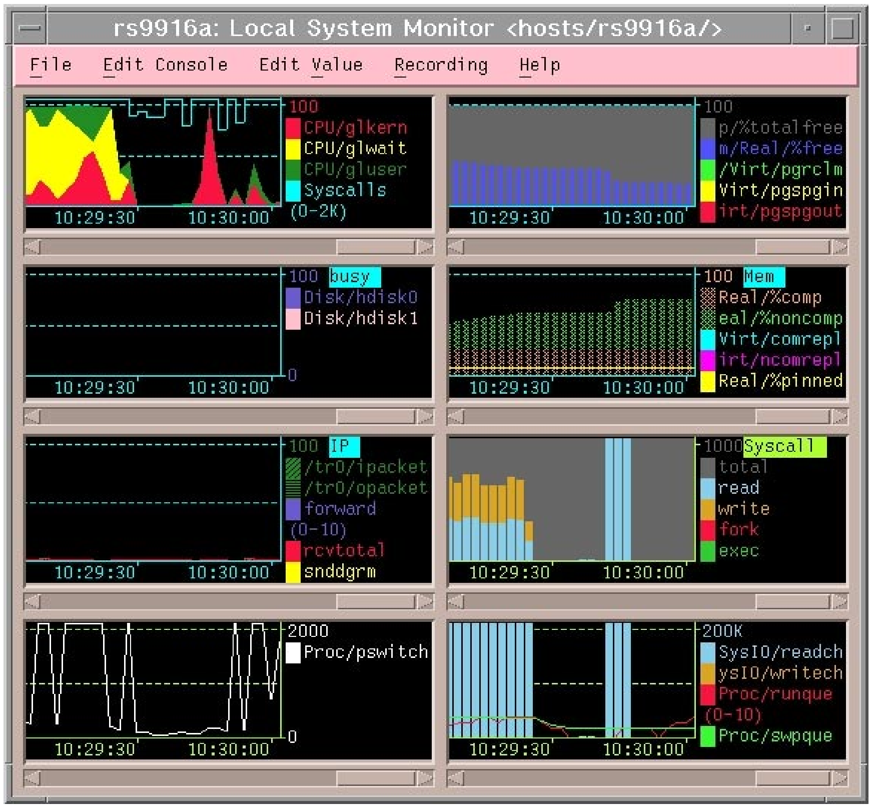


Figure 123. The Monitor window

Figure 123 gives an example of the System Monitor Window. Using the System Monitor utility, it is possible to have a visual image of multiple resources of the local or a remote system. The resources that are monitored may be easily selected and customized. It is also possible to create multiple System Monitor Consoles to accommodate different selections of resources and machines to be monitored. For example, a system administrator could decide to have one console for monitoring memory on a group of machines that he knows usually run memory bound programs and have another console for monitoring process activity on other machines.

It is also possible to track and record the resources while monitoring for a deferred analysis. Figure 124 shows the Combo Style Monitor window.

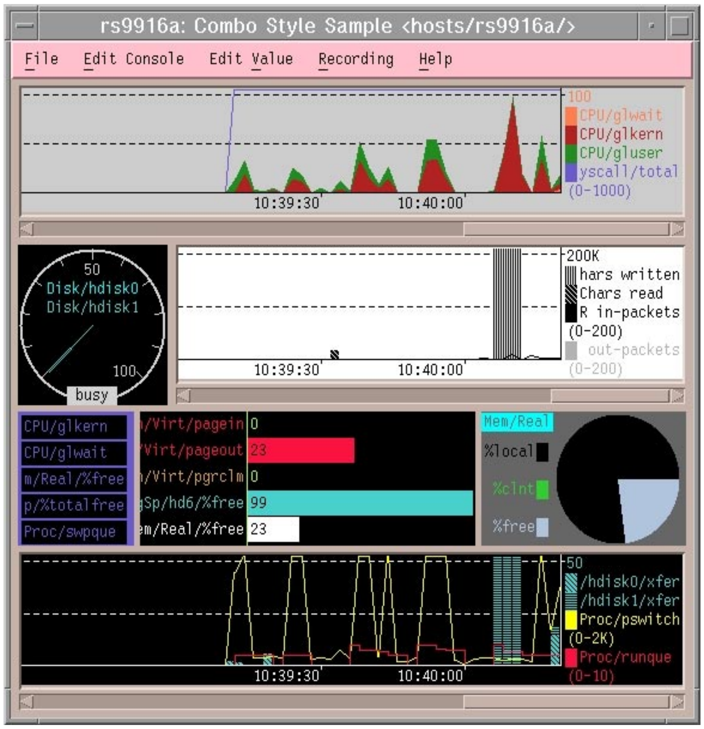


Figure 124. Combo Style monitor window

Other ways to show the selectable system parameters are using the Combo Style Monitor Window, shown in Figure 124, and the Dashboard Style Monitor Window shown in Figure 125 on page 351. In both cases, it is possible to create and customize monitor windows in the same way as described for the Monitor Window shown in Figure 123 on page 349.

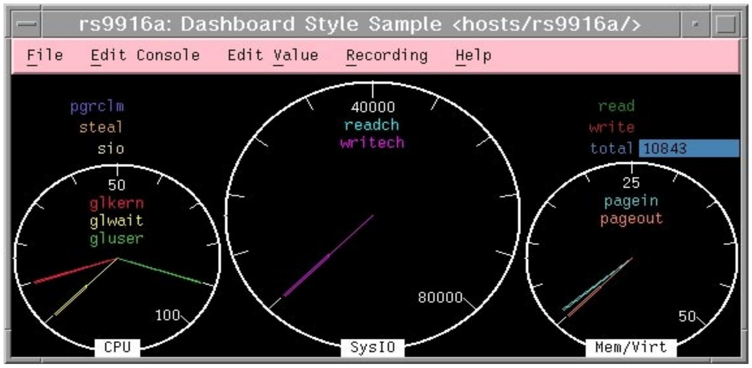
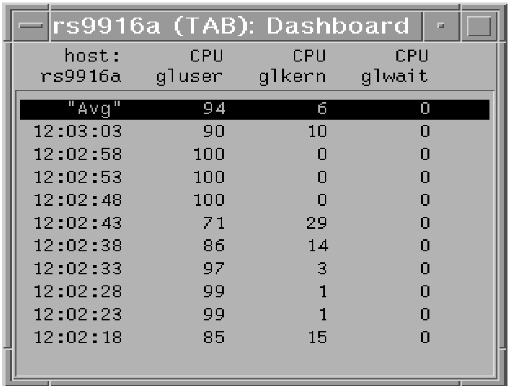


Figure 125. The dashboard style monitor window

With the three Style Monitors just described, it is also possible to select one particular resource from the window and generate a tabular report, as shown in Figure 126.



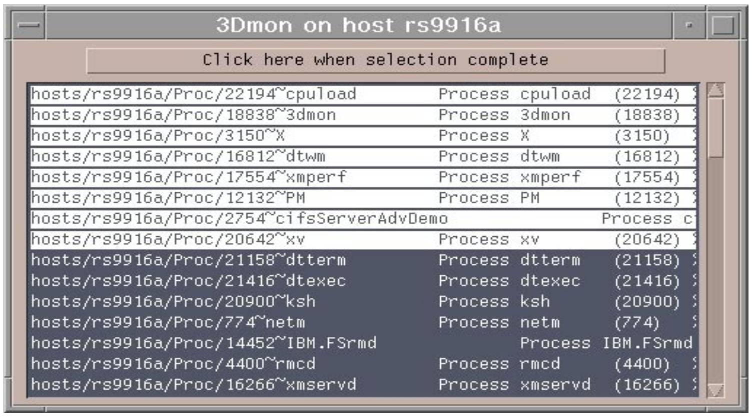
```
   host:      CPU      CPU      CPU
 rs9916a   gluser   glkern   glwait

    "Avg"      94        6        0
 12:03:03      90       10        0
 12:02:58     100        0        0
 12:02:53     100        0        0
 12:02:48     100        0        0
 12:02:43      71       29        0
 12:02:38      86       14        0
 12:02:33      97        3        0
 12:02:28      99        1        0
 12:02:23      99        1        0
 12:02:18      85       15        0
```

Figure 126. A tabular report example

With the tabular report it is easier to have detailed information regarding the selected variable.

One very useful tool for displaying several variables and having them under constant control is the 3dmon tool, which can be accessed from the Utilities item on the main menu bar of the xmperf main window.

Figure 127 on page 352 shows the window that allows you to choose which object to display for monitoring with 3dmon.



```
hosts/rs9916a/Proc/22194~cpuload         Process cpuload   (22194)
hosts/rs9916a/Proc/18838~3dmon           Process 3dmon     (18838)
hosts/rs9916a/Proc/3150~X                Process X         (3150)
hosts/rs9916a/Proc/16812~dtwm            Process dtwm      (16812)
hosts/rs9916a/Proc/17554~xmperf          Process xmperf    (17554)
hosts/rs9916a/Proc/12132~PM              Process PM        (12132)
hosts/rs9916a/Proc/2754~cifsServerAdvDemo                Process c
hosts/rs9916a/Proc/20642~xv              Process xv        (20642)
hosts/rs9916a/Proc/21158~dtterm          Process dtterm    (21158)
hosts/rs9916a/Proc/21416~dtexec          Process dtexec    (21416)
hosts/rs9916a/Proc/20900~ksh             Process ksh       (20900)
hosts/rs9916a/Proc/774~netm              Process netm      (774)
hosts/rs9916a/Proc/14452~IBM.FSrmd               Process IBM.FSrmd
hosts/rs9916a/Proc/4400~rmcd             Process rmcd      (4400)
hosts/rs9916a/Proc/16266~xmservd         Process xmservd   (16266)
```

Figure 127. Selecting objects for 3dmon

In our example, we are going to monitor some processes on the local machine. To do this, we need to select the processes to be monitored. Figure 128 gives an example of the 3dmon display, where the processes selected in Figure 127 on page 352 are displayed.

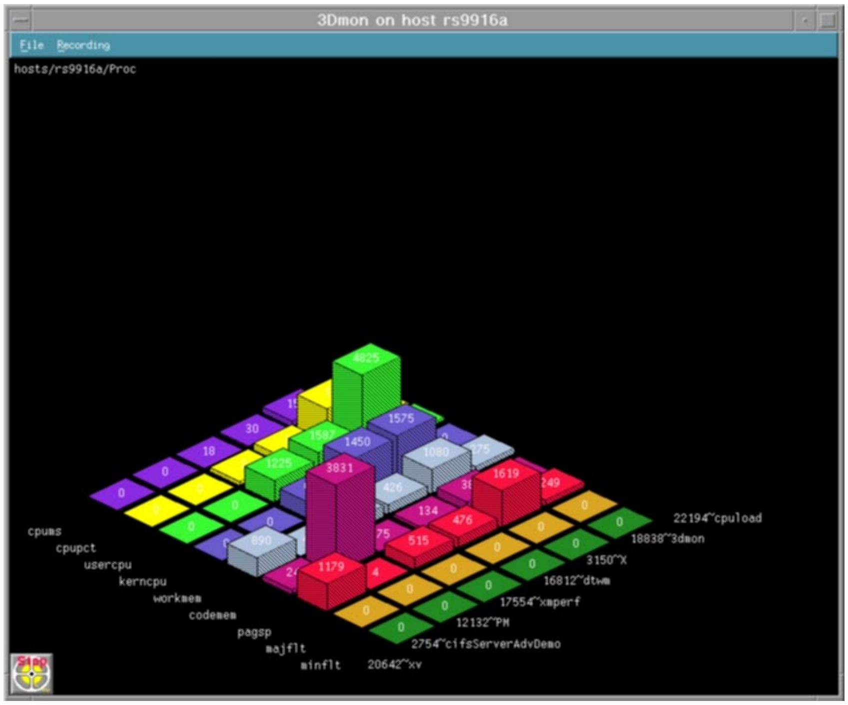


Figure 128. 3dmon

Starting from AIX 5L, Workload Manager (WLM) metrics can be monitored by using xmperf interface as following Figure 129 on page 354. The top instrument displays the CPU metrics such as user, kernel, and wait in a stacked area format. The bottom instrument displays the load of four WLM classed on the system CPU resource in a stacked bar format. There are many formats for the user to select from.

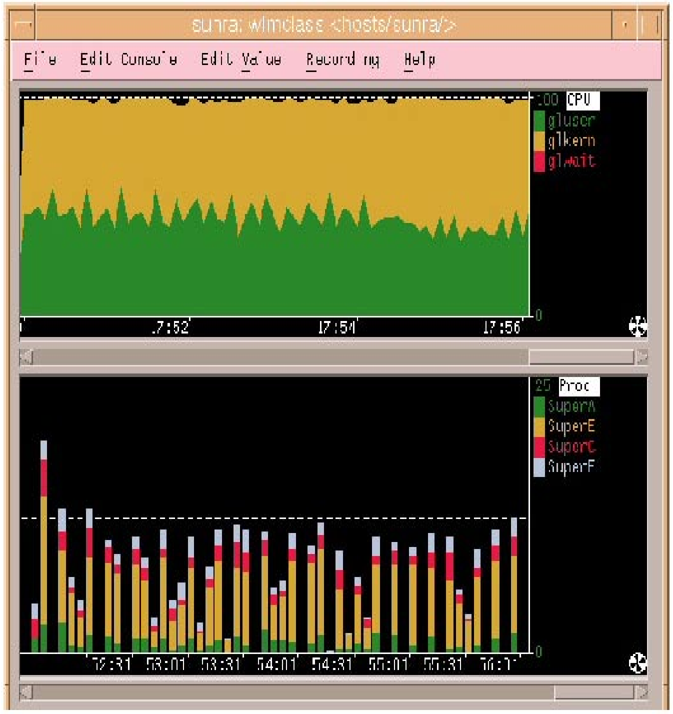


Figure 129. PTX WLM support - CPU Class Display

9.1.6.4 Analysis and control

By providing an umbrella for tools that can be used to analyze performance data and control system resources, the Manager program, xmperf, assists the system administrator in keeping track of available tools and applying them in appropriate ways. This is done through a customizable menu interface. Tools can be added to menus, either with fixed sets of command line arguments to match specific situations or such that the system administrator has an easy way to remember and enter command line arguments in a dialog window. The menus of xmperf are preconfigured to include most of the performance tools shipped as part of the tools option of the Agent component.

Properly customized, xmperf becomes an indispensable repository for tools to analyze and control an AIX system. In addition, the ability to record load scenarios and play them back in graphical windows at any desired speed gives new and improved ways of analyzing performance problems.

Outstanding features for analyzing a recording of performance data are provided by the azizo program and its support programs. Recordings can be produced from the monitoring programs xmperf and 3dmon during monitoring, or they can be created by the xmservd daemon. The xmservd daemon allows for recording with a minimum of overhead. This makes constant recording possible so that you can analyze performance problems after they occur. The 3dplay program is provided to play back recordings created by 3dmon in the same style in which the data was originally displayed. The azizo program creates two window types: Main Window and Main Graph.

Figure 130 gives an example of the Main Graph window.

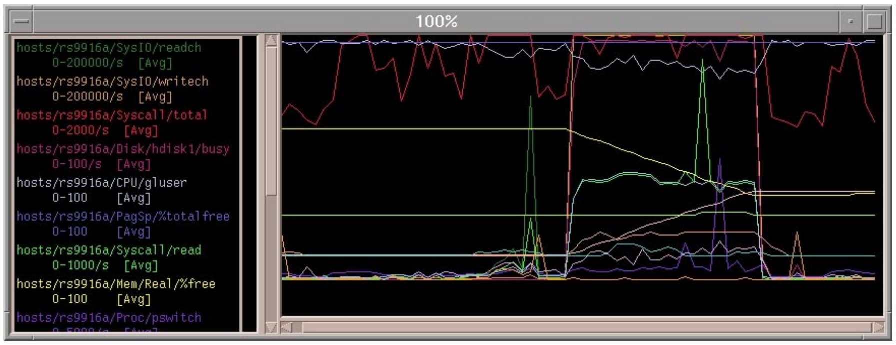

100%
hosts/rs9916a/SysIO/readch
0-200000/s [Avg]
hosts/rs9916a/SysIO/writech
0-200000/s [Avg]
hosts/rs9916a/Syscall/total
0-2000/s [Avg]
hosts/rs9916a/Disk/hdisk1/busy
0-100/s [Avg]
hosts/rs9916a/CPU/gluser
0-100 [Avg]
hosts/rs9916a/PagSp/%totalfree
0-100 [Avg]
hosts/rs9916a/Syscall/read
0-1000/s [Avg]
hosts/rs9916a/Mem/Real/%free
0-100 [Avg]
hosts/rs9916a/Proc/pswitch

Figure 130. The azizo graph window

In the example, we monitored and recorded the system activity using xmperf. Then, we used the azizo program to analyze the output. When azizo reads a recording file, it always displays a top level main graph that covers the entire time interval covered by the recording file. The user can create additional main graphs by zooming in on the top level graph or any zoomed-in main graph. As Figure 130 shows, main graphs contain two sections. To the left is a list of metrics that are included in the graph. The metric names are displayed using the same color that is used to draw the data. To the right is the actual graphical display of the metric data for the time period covered by the graph.

9.1.6.5 Capacity planning

If you can make your system simulate a future load scenario, xmperf can be used to visualize the resulting performance of your system. By simulating the load scenario on systems with more resources (such as more memory or more disks) the result of increasing system resources can be demonstrated.

9.1.6.6 Networked operation

The xmservd data-supplier daemon can provide consumers of performance statistics with a stream of data. The frequency and contents of each packet of performance data are determined by the consumer program. Any consumer program can access performance data from the local host and one or more remote hosts; any data-supplier daemon can supply data to multiple hosts.

In addition to its ability to monitor across a network, PTX also allows for the monitoring of response time to and from nodes in the network itself.

9.1.6.7 Application programming interfaces

Each component comes with its own application programming interface (API). In addition, an API is available for instrumentation of application programs:

- **Agent API** - This is called the System Performance Measurement Interface (SPMI) API. It allows an application program to register custom performance statistics about its own performance or that of some other system component. Once registered, the custom statistics become available to any consumer of statistics, local or remote. Programs that supply custom statistics are called dynamic data-supplier programs. In AIX5L, Work Load Manager (WLM) is added to the SPMI, and the SPMI provides access to hundreds of performance metrics. For each WLM class it includes metrics and associated properties (min, soft max, hard max, target and actual usage). The Agent API also permits applications to access statistics on the local system without using the network interface. Such applications are called local data-consumer programs.
- **Manager API** - This is called the Remote Statistics Interface (RSi) API. It allows an application program to access statistics from remote nodes (or the local host) through a network interface.
- **Application Monitoring API** - This is called the Application Response Management (ARM). This API permits application programs to be instrumented in such a way that the application activity and response time can be monitored from any of the PTX manager programs.

9.1.6.8 SNMP interface

By entering a single keyword in a configuration file, the data-supplier daemon can be told to export all its statistics to a local snmpd SNMP agent. Users of an SNMP manager, such as IBM NetView, see the exported statistical data as an extension of the set of data already available from snmpd.

Note

The SNMP multiplex interface is only available on IBM RISC System/6000 Agents.

9.2 Windows 2000 performance monitoring

As discussed in previous sections, optimizing system performance is a process of locating and isolating system bottlenecks and resolving them. Windows 2000 uses the Microsoft Management Console (MMC) to help the user create, save, and open administrative tools (called MMC consoles) that manage the hardware, software, and network components of the Windows system. MMC is a feature of the Windows 2000 operating system, but it can also be run on Windows NT, Windows 95, and Windows 98 operating systems. MMC does not perform administrative functions but hosts tools that do. For more information about the MMC, refer to Section 8.2.4.3, “Windows 2000 Microsoft Management Console” on page 310.

The Performance Console, found in your Administrative Tools folder, consists of two subtrees:

- System Monitor
- Performance Logs and Alerts

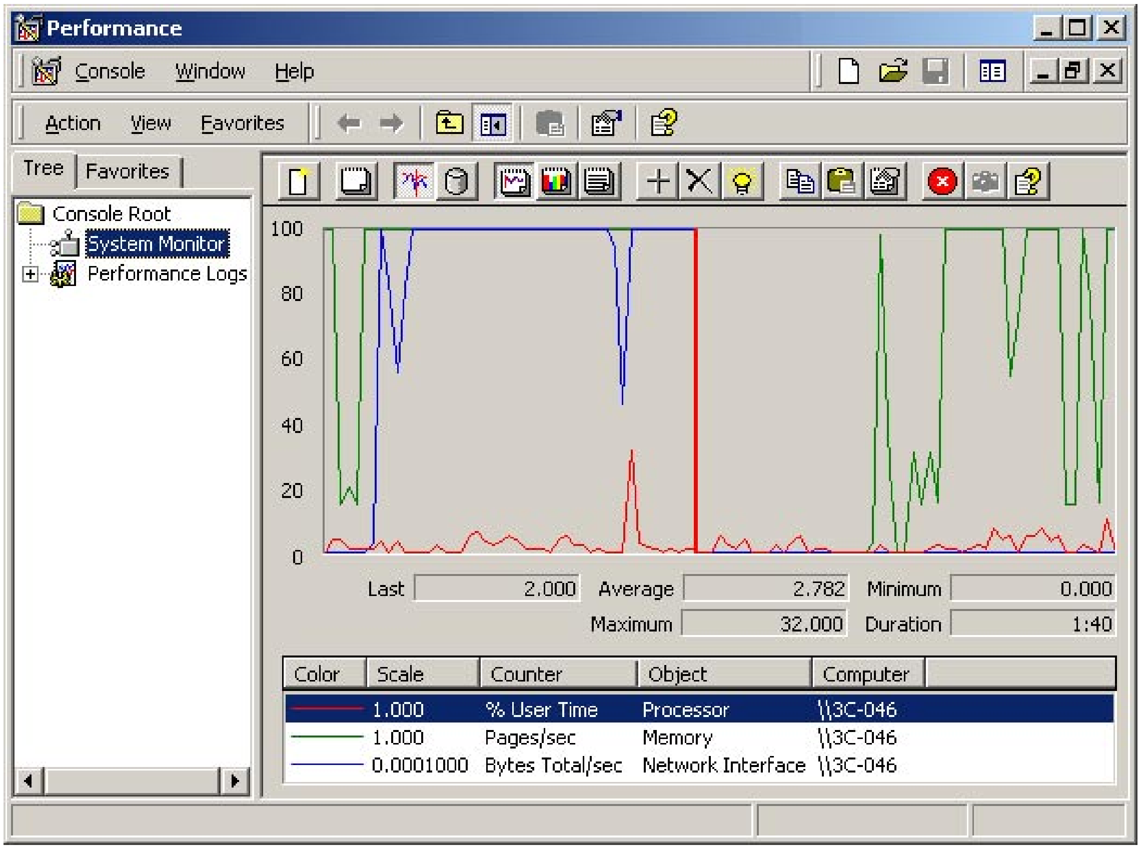
It helps resolve bottlenecks by:

- Providing a view of resource usage (local and remote)
- Logging critical system values
- Sending alerts when important events occur

The following sections give an overview of the Performance Console.

9.2.1 System Monitor

The Windows 2000 System Monitor tracks system resources, or objects, by assigning counters and timers to each resource to be tracked. An object’s counter or timer records the activity level of the object. A counter is expressed as a rate per second, while timers are expressed as the fraction of time that a device is used (shown as a percentage). Figure 131 on page 358 displays the Performance Console. In the example, three objects are tracked: Processor - %User Time, which is the percentage of non-idle processor time spent in user mode, Memory - Pages/Sec, which is the number of pages read from or written to disk to resolve page faults, and Network Interface - Bytes Total/Sec, which is the rate at which data is sent and received on the network interface, in this case a 16Mb Token-Ring card.

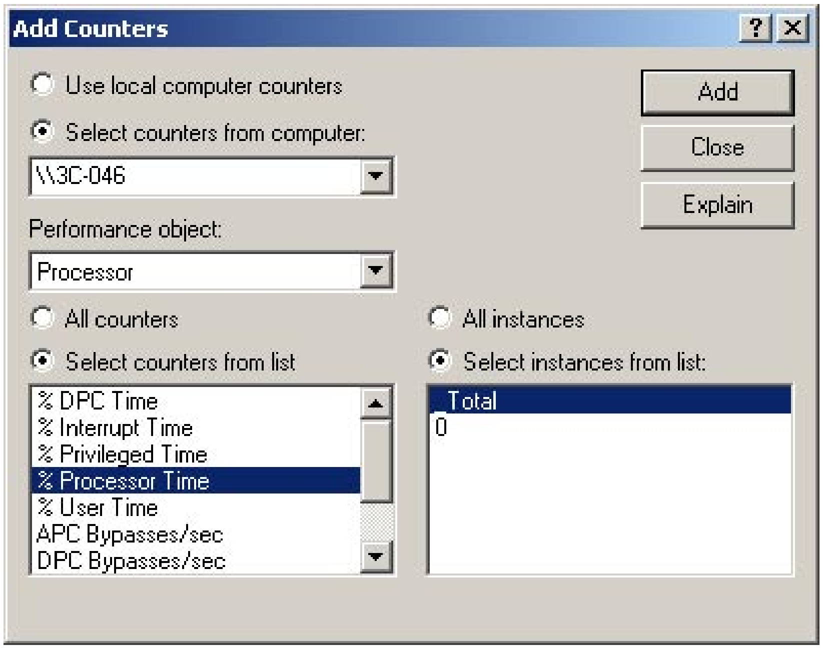


Last 2.000 Average 2.782 Minimum 0.000
Maximum 32.000 Duration 1:40

Color	Scale	Counter	Object	Computer
🟥	1.000	% User Time	Processor	\\3C-046
🟩	1.000	Pages/sec	Memory	\\3C-046
🟦	0.0001000	Bytes Total/sec	Network Interface	\\3C-046

Figure 131. Performance console

Figure 132 on page 359 shows the Add Counters window, which allows you to decide which objects to trace. The components are organized hierarchically inside the window. At the top of the hierarchy is the computer within a domain. Each computer is further broken down into physical components, such as processors, physical disks, and memory. It is possible to browse the performance objects and choose what to monitor simply by selecting the item and clicking the Add button.



○ Use local computer counters

● Select counters from computer:

\\3C-046

Performance object:

Processor

○ All counters

● Select counters from list

% DPC Time
% Interrupt Time
% Privileged Time
% Processor Time
% User Time
APC Bypasses/sec
DPC Bypasses/sec

○ All instances

● Select instances from list:

_Total
0

Figure 132. Add Counters window

A description of the selected counter can be obtained by pressing the Explain button. For most counters, the help is very thorough and informative.

Contrary to Windows NT, disk performance counters are on by default in WIndows 2000. You can still change the disk performance counters' behavior by using the `diskperf.exe` command.

In general, objects are only enabled for system resources that are installed on the machine. For example, if you do not have TCP/IP, you will not be allowed to select counters for collecting TCP/IP statistics on your network.

The system monitor may be used to show values for selected objects both in real time and from collected data. Section 9.2.2, "Performance logs and alerts" on page 360, explains how to collect data do be shown with the System Monitor Console.

9.2.1.1 Report view

The report view is useful for monitoring many counters at the same time. This technique can be used to narrow down the number of variables that might be causing a problem. You can then view the problems most likely to occur using the chart view.

9.2.2 Performance logs and alerts

With Performance Logs and Alerts, it is possible to collect performance data automatically from the local computer or from remote computers and later review logged counter data using the System Monitor or export the data to a spreadsheet file or database for analysis and report generation.

Performance Logs and Alerts offer the following capabilities:

- Performance Logs and Alerts collect data in a comma-separated or tab-separated format for easy import to spreadsheet programs. A binary log-file format is also provided for circular logging or for logging instances, such as threads or processes that may begin after the log starts collecting data. (Circular logging is the process of continuously logging data to a single file and overwriting previous data with new data.)
- Counter data collected by Performance Logs and Alerts can be viewed during collection as well as after collection has stopped.
- Because logging runs as a service, data collection occurs regardless of whether any user is logged on to the computer being monitored or not.
- It is possible to define start and stop times, file names, file sizes, and other parameters for automatic log generation.
- It is possible to set an alert on a counter, thereby, defining that a message be sent, a program run, or a log started when the selected counter's value exceeds or falls below a specified setting.

Similar to System Monitor, Performance Logs and Alerts supports defining performance objects, performance counters and object instances, and setting sampling intervals for monitoring data about hardware resources and system services.

Logs and Alerts may also be started and stopped manually by simply selecting the proper action (start or stop) from the main menu.

The data collected with Performance Logs and Alerts may be displayed from the System Monitor Console by selecting view log file data from the tool bar.

Chapter 10. Networking

Networking is probably one of the most important aspects of an operating system, especially when dealing with a large number of heterogeneous systems. This is often the case in large customer environments. Both AIX 5L and Windows 2000 have introduced major improvements into their networking models. For each product, this chapter describes the network architecture, the network protocols supported, and the communication products.

10.1 Protocols and concepts

In this section, we are going to introduce some protocols and concepts that are common to both AIX and Windows 2000 and have recently been introduced in one or both of these operating systems.

10.1.1 Quality of Service (QoS)

In this chapter, we will introduce Quality of Service, which has recently been added to both AIX 5L and WIndows 2000.

10.1.1.1 Introduction

QoS is a mechanism to improve application response time over the network, and allocate resources, such as bandwidth, to applications. This allocation is determined by giving some applications priority over others. QoS gives administrators control over their networks and, consequently, the ability to provide better service to their customers. For example, a mission-critical application can be guaranteed the resources to complete its transactions within an acceptable period of time.

QoS allows network administrators to use their existing resources efficiently and to guarantee that critical applications receive high-quality service without having to expand as quickly or even over-provision their networks. Deploying QoS means that network administrators can have better control over their networks, reduce costs, and improve customer satisfaction.

There are two QoS models currently under standardization by IETF: Integrated Service (IS) and Differentiated Service (DS).

Integrated Services

Integrated Services (IS) is a dynamic resource reservation model for the Internet defined in RFC1633. Hosts use a signaling protocol called a resource reservation protocol (RSVP) to dynamically request a specific quality of service from the network. An important characteristic of IS is that this signaling is per-flow and reservations are installed per-hop. This means that a host requests resource reservations per logical connection and resource on each hop (routers and hosts) are reserved. Each router reserves resources to classify packets on multiple flows, schedule output of packets based on classes, and so on. This model is well-suited to meeting the dynamically changing needs of applications, but it has significant scaling issues so, it cannot be deployed on routers on backbone networks.

Differentiated Services

Differentiated Services (DS) defined in RFC 2474 removes the per-flow and per-hop scalability issues, replacing them with a simplified mechanism of classifying packets. Rather than a dynamic signaling approach, DS uses six bits in the IP type of service (TOS) byte to separate packets into classes. The particular bit pattern in the IP TOS byte is called the DS codepoint and is used by routers to define the quality of service delivered at that particular hop in much the same way routers do IP forwarding via routing table lookups. The treatment given to a packet with a particular DS codepoint is called a per-hop behavior (PHB) and is administered independently at each network node. When the effects of these individual independent PHBs are concatenated, this results in an end-to-end service. The above two models can be viewed as competing technologies. However, the recent trend is for these two to complement each other. IS is likely to be used in stub networks, and DS is likely to be used in core networks.

10.1.1.2 RSVP

RSVP is an end-to-end layer-3 signaling protocol. RSVP communicates QoS requirements and capabilities to peer-applications and network elements on the data path. RSVP messages also include information about policy, which typically identifies the user requesting resources and the application for which the request is being made.

Traditionally, RSVP signaling was associated with per-flow traffic handling (this is equivalent to each flow having its own queue on the router), but it can also be used with a variety of aggregate traffic-handling mechanisms, such as Diffserv, which takes separate flows and aggregates them into classes. Using RSVP in conjunction with a mechanism, such as Diffserv, should relieve administrators of any concerns they have about scaling problems associated with RSVP.

10.1.1.3 Traffic-handling mechanisms

Traffic-handling mechanisms are a key component of QoS, and they reside in the network's routers and switches. Traffic-handling mechanisms can work in partnership with RSVP, which is a signaling mechanism based in the host.

Diffserv

Diffserv is a layer 3 technology that aggregates different flows of traffic into specific classes of service. This is as if a router had, for example, five queues with different priorities and all traffic had to belong to one of them. A tag in the IP header specifies a packet's priority, and the routers treat it accordingly. Diffserv can work alone or in concert with RSVP. For example, RSVP can be used with Diffserv to enforce admission control into a Diffserv queue and apply policies that determine which users and/or applications are entitled to use resources in the provider's network.

802.1/p

802.1/p is a layer-2 mechanism that allows QoS to be implemented in IEEE 802 technologies, such as Ethernet, FDDI, and token ring. 802.1p defines a field in the layer-2 header of 802 packets that can carry one of eight priority values. Typically, hosts or routers sending traffic into a LAN will mark each transmitted packet with the appropriate priority value. LAN devices, such as switches, bridges, and hubs, are expected to treat the packets accordingly (by making use of underlying queuing mechanisms). The scope of the 802.1p priority mark is limited to the LAN. Once packets are carried off the LAN through a layer-3 device, the 802.1p priority is removed.

ISSLOW

ISSLOW is a technique for improving performance on slow links, such as dialup lines. It is particularly useful when these links carry audio along with video and data. Large data or video packets can cause latency problems for any audio packets that follow them. For example, a 1,500-byte data packet on a 28.8 Kbps modem connection will occupy the link for almost 0.5 seconds. ISSLOW fragments the lower-priority data or video packets so that they never occupy the link for longer than some specified period of time. They are reassembled at the remote end.

Policy and the admission control service

Policy is a set of general rules for handling traffic. For example, a policy might be that SAP traffic should always be given priority over other forms of traffic.

In order for QoS features from various vendors to interoperate correctly, it is necessary to standardize the policy scheme for QoS. An Internet Draft, draft-rajan-policy-qosschema-01.txt, addresses this issue. This document can be found at:

`http://www.internic.net/internet-drafts`

A policy condition is the characteristics of a packet, and a policy action is an action the packet receives when it meets a policy condition.

A policy condition is defined by five characteristics of a packet. They are: Source IP address, source port number, destination IP address, destination port, and protocol type (TCP or UDP). A policy action includes:

- Permission (accept or deny)
- Token bucket parameters defining in-profile traffic
- TOS byte value for in-profile traffic
- TOS byte value for out-of-profile traffic

From an administrator's point of view, a policy is, essentially, configuration parameters to regulate certain types of traffic flow. Policy-based networking applies both IS and DS QoS models.

10.1.2 Path Maximum Transmission Unit (PMTU)

When one IP host has a large amount of data to send to another host, the data is transmitted as a series of IP datagrams. It is usually preferable for these datagrams to be of the largest size that does not require fragmentation anywhere along the path from the source to the destination. This datagram size is referred to as the PMTU, and it is equal to the minimum of the MTU for each hop in the path.

The routing table grows as new PMTU are discovered with very little performance impact on the system. The new route expires after a period of inactivity.

10.1.3 IP multicasting

Both AIX and NT provide Level 2 support for IGMP multicasting (IGMP Version 2), as described in RFC 1112 and RFC 2236. The introduction to RFC 1112 provides a good overall summary of IP multicasting. The text reads:

"IP multicasting is the transmission of an IP datagram to a "host group"-a set of zero or more hosts identified by a single IP destination address. A multicast datagram is delivered to all members of its destination host group with the same *best-effort* reliability as regular unicast IP datagrams; that is, the datagram is not guaranteed to arrive intact to all members of the destination group or in the same order relative to other datagrams."

The membership of a host group is dynamic; that is, hosts may join and leave groups at any time. There are no restrictions on the location or number of members in a host group. A host may be a member of more than one group at a time. A host need not be a member of a group to send datagrams to it.

In addition, a host group may be permanent or transient. A permanent group has a well-known administratively-assigned IP address; it is the address, not the membership of the group, that is permanent. A permanent group may have any number of members at any time, even zero. Those IP multicast addresses that are not reserved for permanent groups are available for dynamic assignment to transient groups that exist only as long as they have members.

Internetwork forwarding of IP multicast datagrams is handled by multicast routers that may be co-resident with, or separate from, Internet gateways. A host transmits an IP multicast datagram as a local network multicast that reaches all immediately-neighboring members of the destination host group. If the datagram has an IP time-to-live greater than 1, the multicast router(s) attached to the local network take responsibility for forwarding it towards all other networks that have members of the destination group. On those other member networks that are reachable within the IP time-to-live, an attached multicast router completes delivery by transmitting the datagram as a local multicast.

10.1.4 TCP selective acknowledgements

Both AIX 5L and Windows 2000 introduce support for TCP Selective Acknowledgements (SACK). Although TCP has a mechanism to recover from the loss of a single segment in a window, in case of multiple segments loss, the sender, generally, has to retransmit not only the lost segments but also the segments received normally.

This causes significant impact on throughput especially when the network is unreliable or congested. To avoid these unnecessary retransmissions, another mechanism called selective acknowledgements (SACK) has been defined in RFC 2018. By enabling SACK, the sender only has to retransmit segments that have really been lost during the transmission.

10.1.4.1 How SACK works

Figure 133 on page 366 illustrates how SACK works.

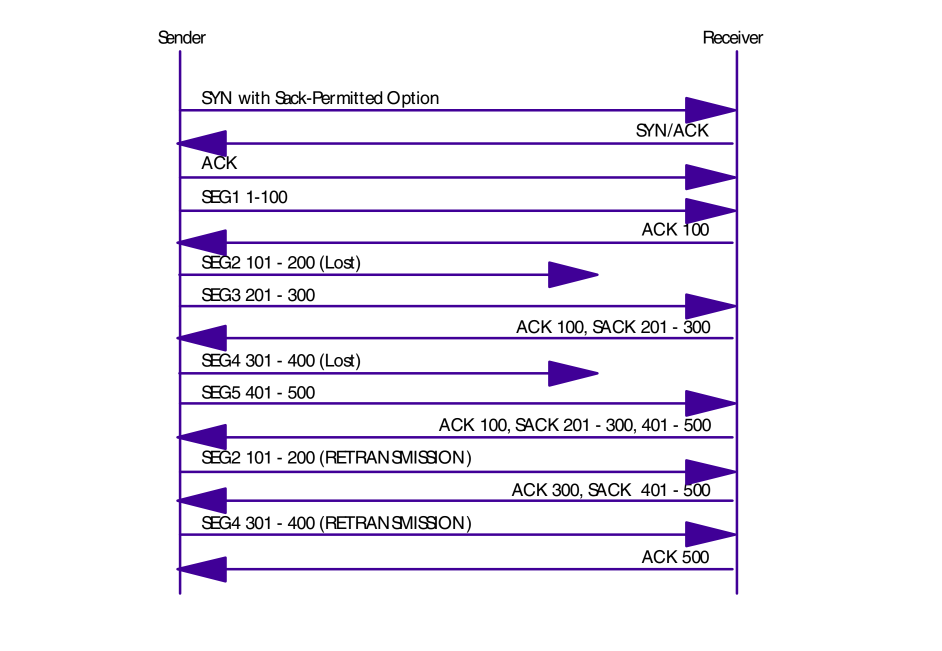


Figure 133. SACK message flow

When establishing the TCP connection, one or both sides of the connection (in this case, only the sender) sends a SYN packet with Sack-Permitted Option. The option means that the sender is ready to receive TCP packets with the SACK Option in the TCP header if the receiver side has implemented the option. The receiver can also send SYN|ACK packets with the Sack-Permitted Option. In this case, the sender also sends its packets with SACK information.

After the connection is established, the sender begins data transmission. In Figure 133, SEG1 is successfully sent, but SEG2 is lost. Since the TCP window on the sender’s side is not filled up yet, the sender sends SEG3, and it is received. At this point, the ACK value sent from the receiver is 100 (SEG1 only) because of the cumulative nature of TCP acknowledgements. But, the packet also includes SACK information indicating that the data within SEG3 (201-300) has been received. Then, the sender knows that it only has to retransmit SEG2 after retransmission time-out.

The above situation (one lost segment) is also resolved by TCP’s fast retransmit/fast recovery mechanism. But, in case of a multiple drop, the sender has no way of knowing the fact and has to wait for the retransmission timer to expire. For example, when SEG4 is lost and SEG5 is received, the ACK value sent back to the sender is still 100. The sender will wait for the retransmission timer to expire and retransmit SEG2, 3, 4, and 5. With SACK, the sender knows that only SEG2 and 4 have been lost; the sender can retransmit these segments.

10.1.5 Security over the network

In this section we will discuss some common concepts regarding security over a public network. In particular, we will introduce the concept of the Virtual Private Network, and we will also focus on a different layer of security mechanisms. Both AIX and Windows 2000 provide support for security over the network, but the protocols supported by them are not exactly the same.

10.1.5.1 Virtual Private Network

The term Virtual Private Network (VPN) is used to describe a set of discrete network connections that communicate securely using encryption through a process called IP tunneling. This allows for a virtually private network over publicly available connections. VPN’s are good implementations for transmitting confidential corporate data or for transmitting personal financial data for eCommerce.

Tunneling is a method of using an internetwork infrastructure to transfer data for one network over another network. Instead of sending a frame as it is produced by the originating node, the tunneling protocol encapsulates the frame in an additional header. The additional header provides routing information so that the encapsulated payload can traverse the intermediate internetwork.

There are mainly two reasons for defining a VPN:

- VPNs allow users working at home or on the road to connect in a secure fashion to a remote corporate server using the routing infrastructure provided by a public internetwork, such as the Internet. Rather than making a long distance call to a corporate or out sourced Network Access Server (NAS), the user calls a local ISP. From the user's perspective, the VPN is a point-to-point connection between the user's computer and a corporate server. The nature of the intermediate internetwork is irrelevant to the user because it appears as if the data is being sent over a dedicated private link.
- VPN technology also allows a corporation to connect to branch offices or to other companies over a public internetwork, such as the Internet, while maintaining secure communications. The VPN connection across the

Internet logically operates as a Wide Area Network (WAN) link between the sites.

Figure 134 shows an example of a Virtual Private Network. The IP packets that flow through the IPSec tunnel are authenticated and, typically, encrypted so that any two entities in intranet A and intranet B can communicate securely.

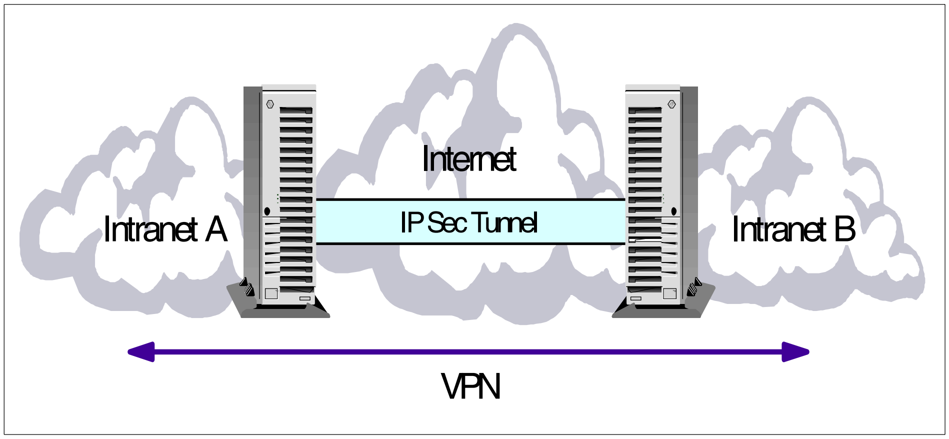


Figure 134. VPN

10.1.5.2 Basic VPN requirements

Typically, when deploying a remote networking solution, an enterprise needs to facilitate controlled access to corporate resources and information. The solution must allow roaming or remote clients to connect to LAN resources, and the solution must allow remote offices to connect to each other to share resources and information (LAN-to-LAN connections). In addition, the solution must ensure the privacy and integrity of data as it traverses the Internet. The same concerns apply in the case of sensitive data traversing a corporate internetwork.

Therefore, a VPN solution should provide at least all of the following:

- **User Authentication** - The solution must verify the user's identity and restrict VPN access to authorized users only. It must also provide audit and accounting records to show who accessed what information and when.
- **Address Management** - The solution must assign a client's address on the private net and ensure that private addresses are kept private.

- **Data Encryption** - Data carried on the public network must be rendered unreadable to unauthorized clients on the network.
- **Key Management** - The solution must generate and refresh encryption keys for the client and the server.
- **Multiprotocol Support** - The solution must handle common protocols used in the public network. These include IP, Internet Packet Exchange (IPX), and so on.

10.1.5.3 Tunneling protocols

Tunneling protocols can be divided into two major categories: Layer 2 tunneling protocols and layer 3 tunneling protocols. Layer 2 protocols correspond to the data-link layer and use frames as their unit of exchange. PPTP and L2TP and Layer 2 Forwarding (L2F) are Layer 2 tunneling protocols; both encapsulate the payload in a PPP frame to be sent across an internetwork. Layer 3 protocols correspond to the Network layer, and use packets. IP-over-IP and IP Security (IPSec) Tunnel Mode are examples of Layer 3 tunneling protocols. These protocols encapsulate IP packets in an additional IP header before sending them across an IP internetwork.

Point-to-Point Protocol (PPP)

Because the Layer 2 protocols depend heavily on the features originally specified for PPP, it is worth examining this protocol more closely. PPP was designed to send data across dial-up or dedicated asynchronous point-to-point connections. PPP encapsulates IP, IPX, and NetBEUI packets within PPP frames and then transmits the PPP-encapsulated packets across a point-to-point link. PPP is used between a dial-up client and an NAS. PPP standard provides two different mechanisms for user authentication to choose from:

- Password Authentication Protocol (PAP) is a simple clear-text authentication scheme
- Challenge-Handshake Authentication Protocol (CHAP); CHAP is an encrypted authentication mechanism that avoids transmission of the actual password on the connection

Point-to-Point Tunneling Protocol (PPTP)

PPTP is a Layer 2 protocol that encapsulates PPP frames in IP datagrams for transmission over an IP internetwork, such as the Internet. PPTP can also be used in private LAN-to-LAN networking.

PPTP is documented in the draft RFC, "Point-to-Point Tunneling Protocol" (pptp-draft-ietf - ppext - pptp - 02.txt). It uses a TCP connection for tunnel maintenance and generic routing encapsulation (GRE) encapsulated PPP frames for tunneled data. The payloads of the encapsulated PPP frames can be encrypted and/or compressed.

Layer 2 Forwarding (L2F)

L2F, a technology proposed by Cisco, is a transmission protocol that allows dial-up access servers to frame dial-up traffic in PPP and transmit it over WAN links to an L2F server (a router). The L2F server then unwraps the packets and injects them into the network. Unlike PPTP and L2TP, L2F has no defined client: In this case, the tunnel is created by a tunnel enabled NAS as soon as the client tries to establish a connection. Such a tunnel, also known as a compulsory tunnel, can be shared by a multiple dial-up client. When a second client dials into the access server (FEP) to reach a destination for which a tunnel already exists, there is no need to create a new instance of the tunnel between the FEP and tunnel server. Instead, the data traffic for the new client is carried over the existing tunnel.

Layer 2 Tunneling Protocol (L2TP)

L2TP is a combination of PPTP and L2F. It is a network protocol that encapsulates PPP frames to be sent over IP, X.25, Frame Relay, or Asynchronous Transfer Mode (ATM) networks. When configured to use IP as its datagram transport, L2TP can be used as a tunneling protocol over the Internet. L2TP can also be used directly over various WAN media, such as Frame Relay, without an IP transport layer.

L2TP is documented in the draft RFC, Layer 2 Tunneling Protocol L2TP (draft-ietf-pppext-l2tp-09.txt). This document was submitted to the IETF in January 1998.

10.1.5.4 Transport layer security

Besides the layer 2 and layer 3 tunneling protocols, many applications today use transport layer security technologies. The most widely-used layer 4 protocols to access secure information over public networks are:

- HTTPS
- SOCKS
- SSL

Transport layer security as provided by SSL/TLS requires that TCP-based applications are written specifically to use these security services.

10.1.5.5 IP Security

In the simplest meaning, IPSec provides ways to authenticate peers and encrypt IP packets along with filtering capabilities. The IP protocol did not have any security mechanisms before IPSec was introduced. Typically, security mechanisms were provided through the use of other transport/application level protocols, such as SSL and Kerberos, or the use of firewalls for packet filtering. By adding robust security mechanisms to the IP protocol layer, most applications can benefit from it without any costs because IP is the most widely used and only network layer protocol in the TCP/IP world.

Basic security services provided by IPSec are similar to those that are provided by other security protocols. They are:

- **Authentication** - IPSec provides a way of authenticating peers. This means that the data originator is really the one the receiver believes they are communicating with.
- **Data integrity** - By using hash functions, IPSec ensures that the data in a packet has not been modified.
- **Encryption** - By encrypting data in a packet, the message in a packet is protected from malicious eavesdroppers.
- **Robustness against replay attacks** - IPSec implements robustness against replay attacks by having sequence number filed in its headers.
- **Access Control** - IPSec provides a packet filter mechanism based on various criteria.

Another important feature of IPSec is that it is independent of any algorithms, although it defines a standard set of algorithms initially. IPSec is designed to work with different kinds of encryption and authentication algorithms. By allowing the selection and replacement of the algorithms, it is possible to change the authentication and encryption methods in the future when better and stronger methods are developed.

There are two security protocols defined for IPsec:

- **Authentication Header** - The purpose of AH is to provide authentication and integrity to IP packets, adding headers to both IPV4 and IPV6 packets. Optionally, it can provide anti-replay attack capability. AH does not have any encryption capabilities; it is often used when no encryption requirements are necessary or where encryption is prohibited.
- **Encapsulating Security Payload (ESP)** - ESP provides authentication, integrity, encryption, and anti-replay attack capability to IP packets. Unlike AH, ESP does not authenticate IP header, and the encryption does not cover IP header and ESP header. In tunnel mode, the IP header is also authenticated and encrypted because the IP header is inside the payload

for the new encapsulating IP packet. ESP is designed for use with symmetric encryption algorithms.

AH and ESP can be used at the same time to increase security.

10.2 AIX networking

This section introduces AIX 5L networking components.

10.2.1 TCP/IP V4

The AIX TCP/IP implementation is based on the BSD 4.3 Reno and BSD 4.4 software release. Figure 135 shows some of the components:

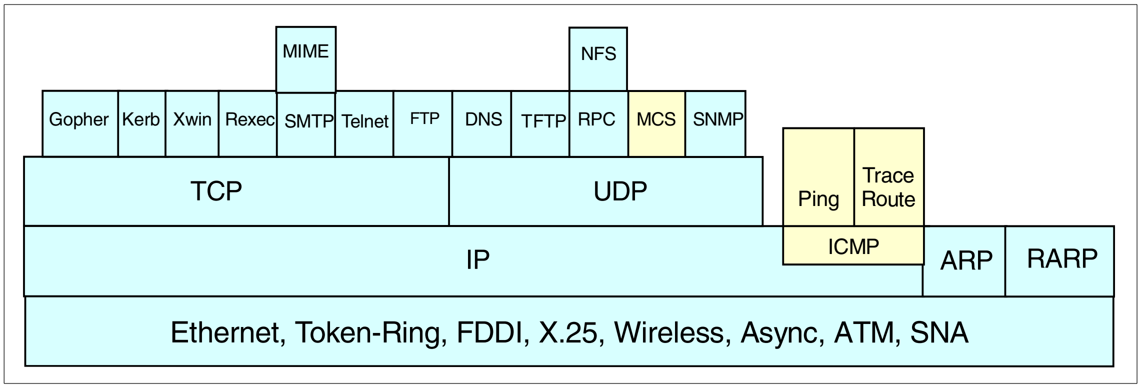


Figure 135. TCP/IP architecture

10.2.1.1 General

Transmission Control Protocol/Internet Protocol (TCP/IP) is a suite of protocols originally developed by the Defense Advanced Research Projects Agency (DARPA). TCP/IP gives the user many commands and facilities with which to transfer files between systems connected to the TCP/IP network, log in to remote systems, print files on remote systems, send electronic mail to remote users, and converse interactively with remote users.

TCP/IP is a protocol that can run on a Local Area Network (LAN) as well as a Wide Area Network (WAN) network; it also has routing capabilities.

10.2.1.2 Supported standards

The AIX for RS/6000 TCP/IP implementation is branded XPG4 (X/Open Portability Guide) and includes the following:

- IEEE POSIX(TM)
  - POSIX 1003.1-1996, includes support for threads option
  - POSIX 1003.2-1993
- X/OPEN(TM)
  - UNIX98 Profile Brand
  - XPG4 Network File Systems (NFS) RFC 1094
  - XPG5 Transport Service (XTI) V2
  - XPG5 Sockets V2
- Terminal Access (TELNET), RFC 856 Binary transmission, RFC 857 Echo OPT, RFC 858 Suppress Go Ahead, RFC 860 Timing Mark, RFC 1073 Window size, RFC 1091 Terminal type, RFC 1123 Internet Hosts Application Support, RFC 974 Mail Routing, RFC 1032 Administrative Guide
- Network Computing System (NCS) 1.5.1
- Domain Name Server, RFC 1033 Administrative Operating Guide, RFC 1034, RFC 1035 Implementation Specification
- Network Management, RFC 1155 MIB (TCP/IP), RFC 1156 MIB (TCP/IP), RFC 1157 SNMP, RFC 1227 (SMUX), RFC 1213 MIB II, RFC 1123 Internet Hosts Application Support, MIL STD
- File Transfer, RFC 1780 FTP, RFC 1350 TFTP, RFC 959 FTP, RFC 822 Message Format
- Name/Finger, RFC 1288
- Time, RFC 868 Time
- Mail, RFC 1123 (SMTP), MIL STD 1781 (SMTP), RFC 821 SMTP, RFC 974 Mail routing
- Service Protocol, MIL STD 1777 TCP, RFC 1122, RFC 793 TCP, RFC 1323, RFC768 UDP, RFC 1122MIL STD 1778 IP, IEEE 802.2 Link, RFC 791 IP, RFC 792 IC Message protocol, RFC 826 Ethernet ARP
- Routing, RFC 888 Stub EGP, RFC 1042 Internet protocol over 802 network, RFC 877 Internet protocol over X.25, RFC 904 EGP format, RFC 950 Subneting, RFC 1122, RFC 1058 Routing

10.2.1.3 Supported physical layers

AIX TCP/IP supports the following physical layers. For some of the different network types, various adapter cards are available and supported. PCI bus, ISA bus, and Microchannel bus configurations are available.

- Token-Ring (TR)
- Ethernet V2 and IEEE 802.3 (EN)
- Fiber Distributed Data Interface (FDDI)
- Asynchronous Transfer Mode (ATM)
- SP2 High Performance Switch (HiPS)
- Fiber Channel Standard (FCS)
- High Performance Parallel Interface (HIPPI)
- Integrated Services Digital Network (ISDN)
- X.25
- Async (by using SLIP or PPP)

In addition, TCP/IP on AIX can also communicate to a mainframe through an S/370, an S/390 Block Multiplexer Channel connection, or an Enterprise System Connection (ESCON) connection.

10.2.1.4 TCP/IP on AIX supported protocols

The following is a list of TCP/IP protocols supported by AIX:

- Core protocols
  - Transport Control Protocol (TCP)
  - User Datagram Protocol (UDP)
  - Internet Control Message Protocol (ICMP)
  - Address Resolution Protocol (ARP)
- TCP/IP connectivity applications
  - TELNET (client and server)
  - FTP (client and server)
  - TFTP (client and server)
  - REXEC (client and server)
  - RSH (client and server)
  - R commands (client and server)
- Network tools
  - `finger/fingerd`
  - `arp`
  - `netstat`

  - `hostname`
  - `host`
  - `ifconfig`
  - `ping`
  - `route`
  - `rwho/rwhod`
  - `traceroute`
  - `tcpdump`
  - `iptrace`
- Network management protocol: Simple Network Management Protocol (SNMP)

10.2.1.5 TCP/IP on AIX additional services

The following is a list of the additional services supported by AIX:

- Network Time Protocol (NTP) Version 3 RFC 1305
- PPP/SLIP
- Simple Mail Transfer Protocol (SMTP)
- Sendmail 8.9.3
- Hyper Text Transfer Protocol (HTTP)
- Network File System (NFS V2 e NFS V3)
- Network Information System (NIS e NIS+)
- NFS Automounter
- WEBNFS
- Distributed File System (DFS)
- DCE Core Services
- Network Computing System (NCS) 1.5.1
- Network Installation Management (NIM)
- License Use Manager (LUM)
- Dynamic Host Configuration Protocol (DHCP)
- Domain Name System (DNS) based on bind 8.1.2
- Dynamic Domain Name System (DDNS)
- X Window (X11R6.3)

- XDMCP (X Window Display Manager Control Protocol)
- routed that implements the Routing Information Protocol (RIP)
- GateD that implements RIP, Exterior Gateway Protocol (EGP), Border
- Gateway Protocol (BGP), the Defense Communications Network
- Local-Network (HELLO) and Open Shortest Path First (OSPF) routing protocol. In addition, the gated daemon supports SNMP.
- Boot Protocol (BOOTP) client and server (bootpd daemon)
- timed that helps synchronizing machines¢ time on a TCP/IP network
- lpr/lpd that allows printing on remote printer

10.2.1.6 Summary of TCP/IP commands

The following is a summary of TCP/IP commands:

- File transfer commands
  - `ftp`
  - `rcp`
  - `tftp`
- Remote login commands
  - `rexec`
  - `rlogin`
  - `rsh` and `remsh`
  - `telnet`, `tn` and `tn3270`
- Status commands
  - `finger` or `f`
  - `host`
  - `ping`
  - `rwho`
  - `ruptime`
  - `whois`
- Remote communication commands
  - `talk`
- Print commands
  - `enq`

- `lpr`

10.2.1.7 Additional commands provided in SP environments

The following are additional commands provided in SP environments:

- distributed shell `dsh`
- `sysctl/sysctld`
- Kerberos
- parallel p* commands
- software update protocol (SUP)
- AMD Automounter

10.2.1.8 Popular Internet protocols available on AIX

The following is a list of most popular public domain protocols available on AIX:

- Network News Transfer Protocol (NNTP) RFC 977
- Internet Relay Chat (IRC)
- Gopher
- Wide Area Information Servers (WAIS) RFC 1625

10.2.2 TCP/IP V6

IP next generation (IPng) is a new version of the Internet Protocol designed as a successor to IP version 4. IPng is assigned IP version number 6 and is formally called IPv6. The next version of TCP/IP is also called IPng (Next Generation) and will be fully supported on AIX. For more information, see RFC 1883 and RFC 1885.

10.2.2.1 IPV6 introduction

IPng was designed to take an evolutionary step from IPv4. It was not a design goal to take a radical step away from IPv4. Functions that work in IPv4 were kept in IPng. Functions that didn’t work were removed. The changes from IPv4 to IPng fall primarily into the following categories:

- **Header Format Simplification** - Some IPv4 header fields have been dropped or made optional to reduce the common-case processing cost of packet handling and to keep the bandwidth cost of the IPng header as low as possible despite the increased size of the addresses. Even though the IPng addresses are four times longer than the IPv4 addresses, the IPng header is only twice the size of the IPv4 header.

- **Improved Support for Options** - Changes in the way IP header options are encoded allows for more efficient forwarding, less stringent limits on the length of options, and greater flexibility for introducing new options in the future.
- **Quality-of-Service Capabilities** - A new capability is added to enable the labeling of packets belonging to particular traffic *flows* for which the sender requests special handling, such as non-default quality of service or *real-time* service.
- **Authentication and Privacy Capabilities** - IPng includes the definition of extensions that provide support for authentication, data integrity, and confidentiality. This is included as a basic element of IPng and will be included in all implementations.

IPng solves the Internet scaling problem, provides a flexible transition mechanism for the current Internet, and was designed to meet the needs of new markets, such as nomadic personal computing devices, networked entertainment, and device control. It does this in an evolutionary way that reduces the risk of architectural problems.

IPng supports large hierarchical addresses that will allow the Internet to continue to grow and provide new routing capabilities not built into IPv4. It has anycast addresses that can be used for policy route selection and scoped multicast addresses that provide improved scalability over IPv4 multicast. It also has local use address mechanisms that provide capability for *plug and play* installation.

Internet Protocol Version 6 (IPv6) was first introduced in AIX version 4.3.0, with support of the host function only. This means that no gateway support is included; so, IPv6 packets cannot be forwarded from one interface to another on the same RS/6000. In AIX version 4.3.2, IPV6 routing is supported.

10.2.2.2 IPV6 128-bit addressing

Here, we are going to provide a brief introduction to the IPV6 addressing mechanism.

As shown in the following example, an IPv6 address is represented by hexadecimal digits separated by colons, where IPv4 addresses are represented by decimal digits separated by dots or full-stops. IPv6 is, therefore, also known as colon-hex addressing, compared to IPv4's dotted-decimal notation.

IPv6 addresses are 128-bit identifiers for interfaces and sets of interfaces. Note that IPv6 refers to interfaces and not to hosts as with IPv4.

There are three conventional forms for representing IPv6 addresses as text strings:

- The preferred form is x:x:x:x:x:x:x:x: where the x's are the hexadecimal values of the eight 16-bit pieces of the address, each separated by a colon.

  Examples are:

  ```
  FEDC:BA98:7654:3210:FEDC:BA98:7654:3210
     1080:0:0:0:8:800:200C:417A
  ```

  Note that it is not necessary to write the leading zeros in an individual field, but there must be at least one numeral in every field (except for the case described next).

- Due to the method used to allocate certain styles of IPv6 addresses, it will be common for addresses to contain long strings of zero bits. To make writing addresses containing zero bits easier, a special syntax is available to compress the zeros. The use of :: (two colons) indicates multiple groups of 16-bits of zeros. Note that the :: can only appear once in an address. The :: can also be used to compress the leading and/or trailing zeros in an address.

  For example, the following addresses:

  ```
  1080:0:0:0:8:800:200C:417A a unicast address
  FF01:0:0:0:0:0:0:43 a multicast address
  0:0:0:0:0:0:0:1 the loopback address
  0:0:0:0:0:0:0:0 the unspecified addresses
  ```

  may be represented as:

  ```
  1080::8:800:200C:417A a unicast address
  FF01::43 a multicast address
  ::1 the loopback address
  :: the unspecified addresses
  ```

- An alternative form that is sometimes more convenient when dealing with a mixed environment of IPv4 and IPv6 nodes is x:x:x:x:x:x:d.d.d.d, where x is the hexadecimal values of the six high-order 16-bit pieces of the address, and d is the decimal values of the four low-order 8-bit pieces of the address (standard IPv4 representation).

  Examples:

  ```
  0:0:0:0:0:0:13.1.68.3
  0:0:0:0:0:FFFF:129.144.52.38
  ```

  or in compressed form:

  ```
  ::13.1.68.3
  ```

```
::FFFF:129.144.52.38
```

Note

FFFF is used to represent addresses of IPv4-only nodes (those that do not support IPv6).

10.2.2.3 Types of IPV6 address

In IPv6, there are three types of addresses:

Unicast

This is an identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address. A unicast address has a particular scope as shown in the following lists:

- link-local
  - Valid only on the local link (that is, only one hop away).
  - Prefix is fe80::/16.
- site-local
  - Valid only at the local site (for example, inside IBM Austin).
  - Prefix is fec0::/16.
- global
  - Valid anywhere in the Internet.
  - Prefix may be allocated from unassigned unicast space.

There are also two special unicast addresses:

- ::/128 (unspecified address).
- ::1/128 (loopback address - Note that, in IPv6, this is only one address, not an entire network).

Multicast

An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to a multicast address is delivered to all interfaces identified by that address. A multicast address is identified by the prefix ff::/8. As with unicast addresses, multicast addresses have a similar scope. This is shown in the following lists:

- Node-local
  - Valid only on the source node (for example, multiple processes listening on a port).

  - Prefix is ff01::/16 or ff11::/16.
- Link-local
  - Valid only on hosts sharing a link with the source node (for example, Neighbor Discovery Protocol [NDP] data).
  - Prefix is ff02::/16 or ff12::/16.
- Site-local
  - Valid only on hosts sharing a site with the source node (for example, multicasts within IBM Austin).
  - Prefix is ff05::/16 or ff15::/16.
- Organization-local
  - Valid only on hosts sharing organization with the source node (for example, multicasts to all of IBM).
  - Prefix is ff08::/16 or ff18::/16.

The 0 or 1 part in these prefixes indicates whether the address is permanently assigned (1) or temporarily assigned (0).

Anycast

This is an identifier for a set of interfaces (typically belonging to different nodes). An anycast address is an address that has a single sender, multiple listeners, and only one responder (normally, the nearest one, according to the routing protocols' measure of distance). An example may be several Web servers listening on an anycast address. When a request is sent to the anycast address, only one responds.

Anycast addresses are indistinguishable from unicast addresses. A unicast address becomes an anycast address when more than one interface is configured with that address.

> **Note**
>
> There are no broadcast addresses in IPv6, their function is superseded by multicast addresses.

10.2.2.4 Additional protocols and functions related to IPV6

There are some additional features that are strictly related to IPng and that are available with AIX; we will now introduce only the most important of these:

- **Internet Control Message Protocol (ICMPv6)** - While IP V4 uses ICMP V4, ICMPv6 is used by IPv6 nodes to report errors encountered in processing packets and to perform other Internet-layer functions, such as diagnostics (ICMPv6 ping) and multicast membership reporting.

- **Neighbor Discovery** - The Neighbor Discovery (ND) protocol for IPv6 is used by nodes (hosts and routers) to determine the link-layer addresses for neighbors known to reside on attached links and maintain per-destination routing tables for active connections. Hosts also use Neighbor Discovery to find neighboring routers that forward packets on their behalf and detect changed link-layer addresses. Neighbor Discovery protocol (NDP) uses the ICMPv6 protocol with a unique message type to achieve the above function. In general terms, the IPv6 Neighbor Discovery protocol corresponds to a combination of the IPv4 protocols Address Resolution Protocol (ARP), ICMP Router Discovery (RDISC), and ICMP Redirect (ICMPv4), but with many improvements over these IPv4 protocols.
- **Stateless Address Autoconfiguration** - IPv6 defines both a stateful and stateless address autoconfiguration mechanism. Stateless autoconfiguration requires no manual configuration of hosts, minimal (if any) configuration of routers, and no additional servers. The stateless mechanism allows a host to generate its own addresses using a combination of locally-available information and information advertised by routers. Routers advertise prefixes that identify the subnet(s) associated with a link while hosts generate an interface-token that uniquely identifies an interface on a subnet. An address is formed by combining the two. In the absence of routers, a host can only generate link-local addresses. However, link-local addresses are sufficient to allow communication among nodes attached to the same link.
- **Tunneling over IP** - The key to a successful IPv6 transition is compatibility with the existing installed base of IPv4 hosts and routers. Maintaining compatibility with IPv4 while deploying IPv6 streamlines the task of transitioning the Internet to IPv6. In most deployment scenarios, the IPv6 routing infrastructure will be built-up over time. While the IPv6 infrastructure is being deployed, the existing IPv4 routing infrastructure can remain functional and can be used to carry IPv6 traffic. Tunneling provides a way to use an existing IPv4 routing infrastructure to carry IPv6 traffic.

10.2.3 GateD Version 6.0

GateD is designed to handle dynamic routing with a routing database built from information exchanged by routing protocols. GateD is a modular software program produced by the Merit GateDaemon Project. It consists of core services, a routing database, and protocol modules supporting multiple routing protocols:

- Routing Information Protocol (RIP)
- Routing Information Protocol Next Generation (RIPng)
- Exterior Gateway Protocol (EGP)
- Border Gateway Protocol (BGP) and BGP4+
- Defense Communications Network Local-Network Protocol (HELLO)
- Open Shortest Path First (OSPF)
- Intermediate System to Intermediate System (IS-IS)
- Internet Control Message Protocol (ICMP) / Router Discovery routing protocols
- Simple Network Management Protocol (SNMP)

GateD was first used to interconnect the NSFNET and emerging regional networks and implement filtered routing based on policy. GateD allows the network administrator to control the import and export of routing information by individual protocol, source and destination autonomous system, source and destination interface, previous hop router, and specific destination address. The network administrator can specify a preference level for each combination of routing information being imported by using a flexible masking capability. Once the preference levels are assigned, GateD decides which route to use independently of the protocols involved. The GateDaemon Consortium presents a formal structure to support and expand the current successful collaborations already in place to develop GateD functionality. Membership is open to all organizations interested in supporting and participating in the development of internetwork routing protocols. Membership is not a prerequisite for licensing of GateD source code. IBM is a member of the GateD consortium. Version 6.0 of GateD from INRIA (Institut National de Recherche en Informatique et en Automatique) is ported from INRIA to AIX 4.3.2 with IPV6 support so that GateD and some of its routing protocols can manipulate IPv6 addresses.

There are three new commands delivered along with the new GateD supporting IPv6 routing.

- The `gdc` command provides an operational user interface for `gated`. The interface is user-oriented for the operation of the GateD routing daemon. It provides supports for:
  - Starting and stopping a GateD daemon
  - Delivering signals to manipulate the GateD daemon

   - Maintenance and syntax checking of configuration files
   - Removal of state dumps and core dumps

  The gdc command can reliably determine the running state of GateD and produces a reliable exit status when errors occur making it advantageous for use in shell scripts that manipulate GateD.

- The ospf_monitor command queries OSPF routers to provide the detailed statistics. It monitors the OSPF gateways.
- The ripquery command sends a RIP request or POLL command, to a RIP gateway to request all the routes known by the gateway. It queries the RIP gateways. This command is used as a tool for debugging gateways.

For more information about the GateD protocol, see the Web site at http://www.gated.merit.edu

10.2.3.1 IPv6 routing functions

The changes for IPv6 routing in Institut National de Recherche en Informatique et en Automatique (INRIA) are merged into the AIX netinet and kernel. Also, some of the routing applications are ported to AIX, specifically, ndpd-router and updates to ndpd-host and ndp. AIX 4.3.2 adds the following IPv6 functions:

- **IPv6 unicast routing support** - This is the normal way a packet is sent to a unicast address and delivered to the interface identified by that address.
- **IPv6 multicast routing support** - This is the way a message is delivered to multiple hosts participating to a multicast group: instead of being routed to only one final destination interface, the packet is routed to several interfaces which declared to join the multicast group
- **IPv6 anycast address support** - In AIX 4.3, the anycast is similar to INRIA. This method involved creating anycast addresses and associating them with a specific interface. This is acceptable because, on incoming packet reception, AIX would talk to all the addresses on the address lists, but, to modify these addresses, a user would have to remember the associated interface. INRIA has moved anycast addresses on to a private list.
- **IPv6 multi-homed link local and site local support** - The reason for adding multi-homed host support is because most routers are multi-homed and need the ability to reference different hosts on different links at the link local and site local levels.

The AIX IPv6 will support the host requirements from the following RFCs:

- **RFC 1883** - Internet Protocol, Version 6 (IPv6) Specification

- **RFC 1884** - IP Version 6 Addressing Architecture
- **RFC 1885** - Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6)
- **RFC 1886** - DNS Extensions to support IP Version 6
- **RFC 1887** - An Architecture for IPv6 Unicast Address Allocation
- **RFC 1970** - Neighbor Discovery for IP Version 6 (IPv6)
- **RFC 1971** - IPv6 Stateless Address Autoconfiguration
- **RFC 1972** - A Method for the Transmission of IPv6 Packets over Ethernet networks
- **RFC 1981** - Path MTU Discovery for IP Version 6
- **RFC 2019** - A Method for the Transmission of IPv6 Packets over FDDI networks

For a complete list of all RFCs and Internet drafts pertaining to IPv6, refer to the Web site:

`http://www.ietf.org/html.charters/ipngwg-charter.html`

This is an evolving list, and AIX conforms to a subset of these.

10.2.4 NFS version 2

Network File System (NFS) support in AIX complies with RFC 1094 developed by Sun Microsystems, Inc. in the late 80s. This version of NFS is also referred to as NFS Version 2. In AIX, NFS includes:

- Network Information System (NIS)
- Remote Procedure Call (RPC) API
- eXternal Data Representation (XDR)

NFS provides the ability to read and write to files that are located physically on another computer system. NFS allows a directory that is in a file system on a remote system to be accessed from the local system as if it was local. To a user, the remote directory and the files in the remote directory appear to be local files. The process of making a remote directory accessible locally involves mounting the directory in a similar way to mounting a local file system.

With NFS, any system can be both a client and a server. A server is a system that is set up to provide access to its local directories and files. A client is a system that is accessing the directories and files on another system. One system can access the files and directories on another system and, at the same time, export its own directories and files to make them available to other systems.

These functions are provided by a combination of the AIX kernel and NFS daemon processes. Multiple NFS daemons are used to service multiple requests at once.

10.2.4.1 Protocols

NFS uses Remote Procedure Calls (RPC) to communicates. RPCs are built on top of the External Data Representation (XDR) protocol which transforms data to a generic format before transmitting and allowing machines with different architectures to exchange information. Figure 136 shows the relationship between the protocols:

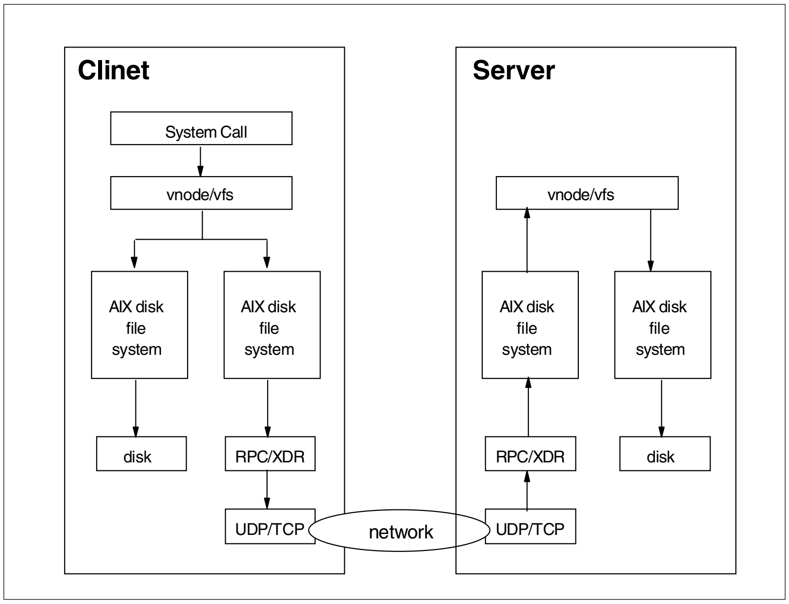


Figure 136. NFS protocol flow

RPC library is a collection of procedures that allows a local (client) process to direct a remote (server) process to execute a procedure call as if the local (client) process had executed the procedure call in its own address space. Because the client and server are two separate processes, they no longer have to exist on the same physical system.

Prior to AIX Version 4.3, the default protocol was UDP for both NFS Version 2 and NFS Version 3. With AIX Version 4.3 and later, TCP is now the default protocol for NFS Version 3, and the client can select different transport protocols for each mount, using the mount command options. You can use a new mount option (proto) to select TCP or UDP. For example:

```
# mount -o proto=tcp
```

10.2.4.2 Limitation on 64-bit kernel

In a heavily loaded environment using high-speed networks (Gigabit Ethernets, for example), NFS with the 64-bit kernel may perform poorly. The symptom of this limitation is socket buffer overflow (as seen by using netstat -s command) while doing file transfers over NFS.

10.2.5 PMTU discovery

RFC1191 provides the mechanism to discover the best PMTU and is supported by AIX. This function was added to AIX Version 4.2.1, but the default is off. With the AIX Version 4.3.3 and later, the default is on. It is possible to enable the TCP PMTU discovery by using the following command:

```
# no -o tcp_pmtu_discover=1
```

10.2.6 NIS

Network Information System (NIS) is a useful tool for administering a large number of systems. The main purpose of NIS is to distribute up-to-date information from AIX files used for user management, system management, and network management. NIS can also be used to distribute information from your own files.

NIS is most commonly used to keep user names, user IDs, passwords, group names, and group IDs consistent across many systems. It provides a means of centrally administrating users, groups, and passwords on a network of machines.

10.2.6.1 NIS domain

An NIS domain is a set of machines that have the same NIS domain name. This domain name is not related to the TCP/IP domain used for address resolution with the DNS/bind protocol. Of course, the NIS domain name can be set to the DNS/bind domain name, but, in general, there will be more systems in the DNS/bind domain than systems sharing the same NIS definitions and having a common NIS master server.

10.2.6.2 NIS maps

NIS does not distribute the actual files containing the data. It uses the information in the files to build an NIS map, which is really a database file created and accessed by NIS clients via remote procedure calls (RPC). NIS uses a database called DBM (Database Management) supplied as standard with AIX.

10.2.6.3 NIS master and slave servers

The information in the NIS maps is kept on a master server, which controls the information. There is only one master server in a single NIS domain. Additional slave servers can hold copies of the information controlled by the NIS master server. The master server automatically updates its slave servers. Slave servers improve performance and availability.

10.2.6.4 NIS security

In general, NIS security is considered weak.

10.2.6.5 NIS Netgroups

In order to assist in managing groups of users and hosts, NIS supports netgroups. Netgroups are definitions that consist of defined users, hosts, and domains.

Netgroup entries have the form:

```
netgroup_name (host_name,user_name,domain_name) (host_name,.....)
```

Netgroups are used as a shorthand way of including host, user, and domain information into NIS maps. Each NIS map only uses the information applicable to it.

10.2.7 Open Network Computing (ONC+)

The ONC+ technology has been licensed from SunSoft and is being included within AIX to meet customer requirements. This technology contains many different functional components; the main ones are: NFS Version 3, NIS+, CacheFS, TIRPC, and AutoFS. Not all of these components are included with this release of AIX. NFS V3 was introduced in AIX Version 4.2.1. CacheFS was introduced in AIX Version 4.3.0; AutoFS was included in AIX 4.3.1, and NIS+ was introduced in AIX 4.3.3.

10.2.7.1 CacheFS

CacheFS is a local disk cache mechanism for NFS clients. It provides the ability for an NFS client to cache file system data on its local disk, thereby, avoiding use of the network and the NFS server when the data is accessed and is not in physical memory. This improves NFS server performance and scalability by reducing server and network load. Designed as a layered file system, CacheFS provides the ability to cache one file system on another. In an NFS environment, CacheFS increases the clients per server ratio, reduces server and network loads, and improves performance for clients, particularly on slow links.

CacheFS is contained in the bos.net.cachefs fileset, which is not automatically installed when installing AIX.

How CacheFS works

After creating a CacheFS file system on a client system, the system administrator specifies which file systems are to be mounted in the cache. When a user on that client attempts to access files that are part of the back file system, those files are placed in the cache. Note that the cache does not get filled until a user requests access to a file or files. Therefore, the initial request to access a file will be at normal NFS speeds, but subsequent accesses to the same file will be at local JFS file system speeds.

10.2.7.2 AutoFS

AutoFS is the component of ONC+ that provides automatic mounting of NFS file systems. It has been included in AIX since AIX 4.3.1. The automounting file system, autofs, mounts file systems when access is requested and unmounts the file system after a few minutes of inactivity, thus, saving the network overhead traffic required to maintain the NFS connection. AutoFS allows you to break the connection when the file system is not being used and restart it again automatically the next time access is desired.

How AutoFS Works

AutoFS is a client-side service. It is implemented using three components to accomplish automatic mounts. The components are:

- The `automount` command
- The autofs kernel extension
- The `automountd` daemon

The `automount` command is called at system startup time from /etc/rc.tcpip. It loads the autofs kernel extension (if it is not already loaded) and reads the master map information from the file /etc/auto_master. The `automount` command then passes the information it read from the master map to the autofs kernel extension. It then starts the automountd daemon and terminates.

The autofs kernel extension reads the information passed to it from the `automount` command and maintains an internal table of the autofs mounts. These autofs mounts are not automatically mounted at startup time. They are points under which file systems can be mounted in the future.

When a client attempts to access a file system that is not presently mounted, the autofs kernel extension intercepts the request and gets the automountd daemon to mount the requested directory. The automountd daemon locates the directory, mounts it within autofs, and replies. Upon receiving the reply, autofs allows the waiting request to proceed. Subsequent references to the mount are redirected by the autofs. No further participation is required by automountd.

With this implementation of automatic mounting, the automountd daemon is completely independent of the `automount` command. Because of this separation, it is possible to add, delete, or change map information without first having to stop and start the automountd daemon process. Once the filesystem is mounted, further access does not require any action from automountd.

10.2.7.3 NFS server performance enhancement

The NFS server performance of AIX 4.3.2 is enhanced with the implementation of a vnode cache in the JFS component of the kernel. The cache enables the NFS server code to translate an NFS file handle to a local vnode structure more efficiently than previous versions of AIX. As a result, the NFS server code spends less time holding a VFS lock word, which, in turn, increases the available throughput of the NFS server.

10.2.7.4 NIS+

NIS+ is a new feature of AIX 4.3.3, which is a part of SUN's NIS+ Version 3 software.

NIS+ expands the network name service provided by NIS. NIS+ enables the storage of information about workstation addresses, security information, mail information, ethernet interfaces, and network services in central locations where all workstations on a network can access it. This configuration of network information is referred to as the NIS+ name space.

The NIS+ name space is hierarchical and can be configured to conform to the logical hierarchy of an organization. An NIS+ name space can be divided into multiple domains, each of which can be administered autonomously. Clients may have access to information in other domains as well as their own if they have the appropriate permissions.

NIS+ uses a client-server model to store and have access to the information contained in an NIS+ name space. Each domain is supported by a set of servers. The principal server is called the master server and the backup servers are called replicas. The network information is stored in standard NIS+ tables in an internal NIS+ database. Both master and replica servers run NIS+ server software and both maintain copies of NIS+ tables. Changes made to the NIS+ data on the master server are incrementally and automatically propagated to the replicas.

NIS+ includes a sophisticated security system to protect the structure of the name space and its information. It uses authentication and authorization to verify whether a client's request for information should be filled. Authentication determines whether the information requester is a valid user on the network. Authorization determines whether a particular user is allowed to have or modify the requested information.

Differences between NIS and NIS+

NIS+ differs from NIS in several ways. It has many new features, and the terminology for similar concepts is different. Table 20 gives an overview of the major differences between NIS and NIS+.

Table 20. NIS and NIS+ differences

NIS	NIS+
Machine name and user's name can be the same.	Machine name and user names must be unique. Furthermore, you cannot have a dot (.) in your machine or user name.
Domains are flat; there is no hierarchy.	Domains are hierarchical; data is stored in different levels in the name space.
Names and commands are case sensitive.	Names and commands are not case-sensitive.
Data is stored in two-column maps.	Data is stored in multi-column tables.
Uses no authentication.	Uses DES authentication.
An NIS record has a maximum size of 1024 bytes. This limitation applies to all NIS map files. For example, a list of users in a group can contain a maximum of 1024 characters in single-byte character set file format.	The NIS+ record has no limit.
Provides single choice of network information source.	Client chooses information source: NIS, NIS+, DNS, or local /etc files.

NIS	NIS+
Updates are delayed for batch propagation.	Incremental updates are propagated immediately.

NIS+ is designed to replace NIS, not enhance it. NIS was intended to address the administration requirements of smaller client-server computing networks. Typically, NIS works best in environments with no more than a few hundred clients, a few multipurpose servers, only a few remote sites, and trusted users.

The size and complexity of modern client-server networks require new autonomous administration practices. NIS+ was designed to meet the requirements of networks that typically range from 100-10,000 multi-vendor clients supported by 10-100 specialized servers located in sites throughout the world. In addition, the information they store can change rapidly.

Because more distributed networks require scalability and decentralized administration, the NIS+ name space was designed with hierarchical domains that may be administered independently.

Although this division into domains makes administration more autonomous and growth easier to manage, it does not make information harder to access. Clients have the same access to information in other domains as they would have had under one umbrella domain. A domain can even be administered from within another domain.

NIS compatibility mode

NIS-compatibility mode enables an NIS+ server to answer requests from NIS clients while continuing to answer requests from NIS+ clients. NIS+ does this by providing two service interfaces. One responds to NIS+ client requests, while the other responds to NIS client requests.

This mode does not require any additional setup or changes to NIS clients unaware that the server responding is not an NIS server. However, there are some differences including the fact that the NIS+ server running in NIS-compatibility mode does not support the ypupdate and ypxfr protocols and, thus, it cannot be used as a replica or master NIS server. Instructions for setting up a server in NIS-compatibility mode are slightly different than those used to set up a standard NIS+ server, and NIS-compatibility mode has security implications for tables in the NIS+ name space. Since the NIS client software does not have the ability to provide the credentials that NIS+ servers expect from NIS+ clients, all their requests end up classified as unauthenticated. Therefore, to allow NIS clients to access information in NIS+

tables, those tables must provide access rights to unauthenticated requests. This is handled automatically by the utilities used to set up a server in NIS-compatibility mode.

NIS+ structure and concepts

The NIS+ name space is the way information is stored by NIS+. The name space can be arranged in a variety of ways to suit the needs of an organization. For example, if an organization had three divisions, its NIS+ name space would likely be divided into three parts, one for each division. Each part would store information about the users, workstations, and network services in its division, but the parts could easily communicate with each other. Such an arrangement would make information easier for users to access and for administrators to maintain. Without entering too deeply into details, let us simply assume that the three basic components that are used by NIS+ to structure this information are called directories (they represent the skeleton of the name space), tables (the objects that contain information), and groups (collections of NIS+ principals used to allow access to NIS+ tables).

To help you understand what an NIS+ name space is, let us say that it may resemble a traditional UNIX file system but with some important differences as described in Table 21 on page 393.

Table 21. NIS+ name space and UNX file system structural comparison

NIM name space	UNIX file system
Uses tables and groups.	Uses files.
Administered only through NIS+ commands.	Administered with UNIX standard file system commands.
the names of NIS+ name space objects are separated by dots (wiz.com.).	The names of AIX file system components are separated by slashes (/usr/bin).
The root of NIM+ name space is reached by stepping from left to right (sales.witz.com.).	The root of AIX filesystem is reached by stepping from right to left (/usr/src/file1).
Because NIS+ object names are structured from left to right, a fully qualified name always ends in a dot. Any NIS+ object ending in a dot is assumed to be a fully qualified name. NIS+ object names that do not end in a dot are assumed to be relative names.	A fully qualified AIX object start with a slash (root) and ends with the name of the object itself.

A fully-qualified NIS+ domain name is formed from left to right starting with the local domain and ending with the root domain:

```
wiz.com.
sales.wiz.com.
intl.sales.wiz.com.
```

The first line shows the name of the root domain. The root domain must always have at least two labels and must end in a dot.

NIS+ domain's components

Every NIS+ domain is supported by a set of NIS+ servers. The servers store the domain's directories, groups, and tables and answer requests for access from users.

Two types of servers support an NIS+ domain: A master and its replicas. The master server of the root domain is called the root master server. A name space has only one root master server. The master servers of other domains are simply called master servers. Likewise, there are root replica servers and regular replica servers.

Both master and replica servers store NIS+ tables and answer client requests. The master, however, stores the master copy of a domain's tables. The replicas store only duplicates. The administrator loads information into the tables in the master server, and the master server propagates it to the replica servers.

This arrangement has two benefits. First, it avoids conflicts between tables because only one set of master tables exists; the tables stored by the replicas are only copies of the masters. Second, it makes the NIS+ service much more available. If either the master or a replica is down, another server can act as a backup and handle the requests for service.

Similarly, an NIS+ client is a workstation that has been set up to receive NIS+ service. Setting up an NIS+ client consists of establishing security credentials, making it a member of the proper NIS+ groups, verifying its home domain, and, finally, running the NIS+ initialization script. An NIS+ client can access any part of the name space if it has been authenticated and granted the proper permissions. Anyway, a client belongs to only one domain, which is referred to as its home domain.

10.2.7.5 NFS V3

NFS V3 has been included in AIX since Version 4.2.1 and brings many improvements for both the NFS client and server. The most notable of those improvements is that it allows the NFS client to request an asynchronous write and commit sequence for writing file data. This new feature allows for faster file writes to the NFS server. With the NFS Version 2 protocol, the NFS server must write file data to disk before responding to the NFS client. This is not required with the use of the NFS Version 3 asynchronous write request.

NFS Version 2 limited the size of READ and WRITE requests to 8 KB. The NFS Version 3 protocol relaxes the transfer size for READs and WRITEs. The AIX implementation, such as most in the industry, offers a 32 KB READ and WRITE size for both client and server. For example, with NFS Version 2 the reading of a 128 KB file would require the NFS client to send 16 individual remote procedure calls to the NFS server. With NFS Version 3, the same file could be read with four remote procedure calls.

- The NFS Version 3 protocol was developed with the ability to access files larger than 2 GB in size. The AIX NFS client and server take advantage of this ability and provide access to files greater than 2 GB in size.

Besides these major enhancements, at least two other functional changes have been introduced into AIX since Version 4.2.1, both in the NFS client and server:

- **NFS over TCP** - The NFS client and server can use tcp instead of udp for communication
- **Multithreaded NFS server** - NFS client and server daemons are implemented in AIX 4.2.1 with the use of AIX multithreading support. The NFS server daemon, nfsd, has been a multiprocess implementation in the past. With the new multi-threaded NFS server load, balancing the server becomes much easier. NFS server threads are created and destroyed on demand as the incoming NFS client requests increase and decrease. The NFS client also takes advantage of the multi-threaded approach to provide a well-balanced resource approach to reading and writing files.

10.2.8 WebNFS

WebNFS is an extension of NFS V3 to the Internet. WebNFS allows users to access files and images from a WebNFS server through Internet connections (modem, slow speed lines) as if they were local. When accessing a file, the file is automatically mounted over the network. WebNFS is able to recover from the drop of a line. WebNFS is faster than HTTP when displaying files and graphics.

It also supports the URL NFS (for example, `nfs://servername/filename`). With WebNFS, Web editors can work directly with the original files without downloading, editing, and sending the file back to the Web server.

10.2.9 Name resolution

This section covers the different mechanisms existing in AIX to resolve names of systems on the TCP/IP network to IP addresses and vice versa.

10.2.9.1 Hosts file

The first way to resolve host names to IP addresses in a TCP/IP network is to maintain an /etc/hosts file on each machine participating in the network. This file contains a list of IP addresses and their corresponding host names. While the mechanism is relatively simple, it is a burden on large networks because the /etc/hosts file must be updated on each machine every time a machine is added to or removed from the network. It is, however, fine for small networks.

10.2.9.2 NIS

NIS allows one to maintain a centralized copy of the hosts file. This simplifies the update of the hosts file since only one copy needs to be maintained. When the network becomes quite large, managing a large number of systems via NIS is not very elegant. If hosts are added to the name space at various locations, NIS and the hosts file is an unmanageable solution. NIS+ or DNS are a better choice.

For a more complete discussion of NIS, see section 10.2.6, “NIS” on page 387.

10.2.9.3 NIS+

Like NIS, NIS+ allows the central administration of the host database. Unlike NIS, NIS+ is much more scalable, given its hierarchical structure, and may be used for more distributed environments than NIS.

Besides NIS+ not only allows central management of IP addresses of machines in the network, but it enables system administrators to store information about client addresses, security information, mail information, network interfaces, and network services in central locations where all the clients on a network can access it. For further details about NIS+, see section 10.2.7.4, “NIS+” on page 390.

10.2.9.4 DNS and DDNS

The Domain Name System (DNS) is the way that host names are organized in a TCP/IP network. If you have a site with many systems, DNS is used to delegate the responsibility for naming systems to other people or sites. This reduces your administration workload by only having to update one server if you want to change the address of a system.

The domain system is not limited to finding Internet addresses. Each domain name is a node in a database. The node can have records that define a number of different properties. Examples are Internet address, computer type, and a list of services provided by a computer. A program can ask for a specific piece of information or for all information about a given name. It is possible for a node in the database to be marked as an *alias* (or nickname) for another node. It is also possible to use the domain system to store information about users, mailing lists, or other objects.

DNS is strictly integrated with DHCP, which allows automatic registration of IP addresses into the DNS of clients as soon as they are provided by the DHCP server (DDNS). For more details about DNS, see section 10.2.10, “Domain Naming System (DNS)” on page 397, and, for a more complete discussion of DDNS, refer to section 10.2.12, “Dynamic Domain Name System (DDNS)” on page 403.

10.2.10 Domain Naming System (DNS)

The Domain Naming System (DNS) is a method for distribution of a large database of IP addresses, hostnames, and other record data across administrative areas. The end result is a distributed database maintained in sections by authorized administrators per domain.

Similar to other UNIX platforms, AIX offers the BIND DNS server. BIND, or the Berkeley Internet Name Daemon, is a DNS server implementation provided by the Internet Software Consortium. It has become the standard for DNS server implementations and a benchmark for DNS server intercompatibility.

10.2.10.1 Domain structure

A host name is the name of a machine. The host name is usually attached to the left of the domain name. The result is a host’s domain name. Domain names reflect the domain hierarchy. Domain names are written from the most specific (a host name) to the least specific (a top-level domain), from left to right, with each part of the domain name separated by a dot. A fully-qualified domain name (FQDN) starts with a specific host and ends with a top-level domain followed by the root domain (the dot, “.”). www.xyx.com. is the FQDN of workstation www in the xyz domain of the com top level domain. A domain is part of the name space, and it may cover several zones.

10.2.10.2 DNS zone

A zone is part of the name space. If a nameserver is listed at the InterNIC or at a higher level nameserver) as authoritative for part of the name space, and it has full data on that part of the name space, it is authoritative for that zone.

10.2.10.3 Types of Domain name servers

Servers do not really have types. A server can be primary for some zones and secondary for others. However, a server cannot be primary and secondary for the same zone. Additionally, a server may serve no zones and just answer queries via its cache.

10.2.10.4 Primary Domain Server

There is only one primary server per zone. The data is always loaded from a file.

10.2.10.5 Secondary Domain Server

On the secondary domain server, the data is always transferred from a primary domain server and stored in a local file. It checks every refresh period with the primary, looking for changes. There is an unlimited number of secondaries per zone. In general, there are one or two on each subnet to split the load and provide the availability of the service.

10.2.10.6 Caching only Domain Server

All servers are caching servers. This means that the server caches the information that it receives for use until the data expires. A Caching Only Server is a server that is not authoritative for any zone. This server services queries and asks other servers, which have the authority, for the necessary information. Al l servers keep data in their caches until the data expires, based on a TTL (Time To Live) field that is maintained for all resource records.

10.2.10.7 Forwarder server

Any server can make use of forwarders. A forwarder is another server, capable of processing recursive queries, that is willing to try to resolve queries on behalf of other systems.

10.2.10.8 Slave servers

Slave mode is used if the use of forwarders is the only possible way to resolve queries due to a lack of full net access or if you wish to prevent the name server from using other than the listed forwarders.

10.2.10.9 AIX implementation of DNS and BIND

AIX 4.3.2 and higher Version incorporate IBM DNS value-added functions to the latest level of BIND, Version 8.1.2. This involves adding IBM secure dynamic DNS update protocol and incremental zone transfers to BIND 8.1.2 as well as extending this BIND’s NOTIFY ability and parameter configuration.

The following are the new functions provided by Bind Version 8.1.2:

- **Secure dynamic DNS updates** - Currently, BIND offers only the unsecured RFC 2136 update protocol. This is an insufficient offering for customers desiring to implement a dynamic DNS environment in their networks. The secure update protocol is added as the solution for RFC 2136's insecure shortcomings and to provide backward compatibility with current AIX dynamic DNS customers.
- **Incremental zone transfer** - Implement the RFC 1995 Incremental Zone Transfer protocol. This protocol defines a method through which secondary DNS servers can update their existing zone data to incorporate all the cumulative changes to the primary zone since the last transfer. This protocol supersedes the performance of ordinary zone transfers by limiting the amount of network traffic between primary and secondary DNS servers and the subsequent computation time of incorporating an entirely new zone. The protocol ensures that incremental zone transfers can be sent to indicate changes from both dynamic updates and zones changed on disk (those reincorporated through a refresh signal or server restart).
- **Notify** - Implementing the RFC 1996 Notify process. This is a method by which the primary DNS server can indicate to its secondary nameservers that zone data has been updated. This decreases the time periods in which a secondary DNS server will have data out of synchronization with its primary DNS server.
- **File Conversion Utility** - Extending the configuration file conversion utility to support IBM functional additions to previous BIND releases. This involves mapping the dynamic keywords of previous named.boot files to a functional equivalent in the named.conf configuration file.
- Proprietary protocol for secure updates of dynamic notify dynamic zones.

  Bind 4.9.3 is available using named4. Bind 8.1.2 uses named8.

10.2.10.10 DNS Support for Load Balancing

Around 1986, a number of different schemes started surfacing as hacks to the Berkeley Internet Name Domain server (BIND) distribution. Probably the most widely distributed of these were the “Shuffle Address” (SA) modifications by Bryan Beecher, or possibly Marshall Rose’s “Round Robin” code. RFC 1794 contains more detailed information on this topic.

IBM offers SecureWay Network Dispatcher, which is included in IBM WebSphere Performance Pack V3 (5639-I29), which supports load balancing for DNS server.

10.2.10.11 MX - Mail eXchange

The domain system is particularly important for handling computer mail. There are entry types to define what computer handles mail for a given name, to specify where an individual is to receive mail, and to define mailing lists. Mail eXchange records, MX, are used to specify a list of hosts that are configured to receive mail sent to this domain name. Every host name that receives mail should have an MX since, if one is not found at the time mail is being delivered, an MX will be *imputed* with a cost of 0 and a destination of the host itself. If you want a host to receive its own mail, you should create an MX for your host's name pointing at your host's name. It is better to have this be explicit than to let it be imputed by remote mailers.

10.2.10.12 DCE intercell communication

The domain system is also used for Distributed Computing Environment (DCE) intercell communication. A DCE sub-type of record is usually accompanied by a TXT record for other information specifying other details to be used in accessing the DCE cell. RFC 1183 contains more detailed information on the use of this record type.

10.2.11 Dynamic Host Configuration Protocol (DHCP)

The Dynamic Host Configuration Protocol is an IETF-approved standard. DHCP implementation in AIX adheres to IETF RFC 1533, RFC 1534, RFC 1541, and RFC 1542 and is a full implementation of these standards. DHCP is an application-layer protocol that allows a machine on the network to automatically query an IP address and other network configuration parameters, such as host name, domain name, DNS servers, subnet mask, default gateways, and so on. The DHCP server can dynamically assign all the information needed to the client. That information does not need to be entered on the DHCP client. All network information is assigned when the client boots up for the first time.

10.2.11.1 DHCP is based on BOOTP

The DHCP protocol is actually based on the BOOTP protocol. The BOOTP protocol is the protocol used by diskless workstations, routers, and terminal concentrators to obtain their network information from a boot server. These BOOTP clients usually do not know their IP addresses (they only know their physical network address), and, if they do not know the BOOTP server address, they temporarily use the IP address 0.0.0.0 to contact the BOOTP server by issuing a limited broadcast. The BOOTP server will then send the required network information back to the client.

10.2.11.2 DHCP Simplifies Network Management

The DHCP mechanism can greatly simplify network administration since a user does not have to request an IP address and associated network information from the network administrator. Also, the administrator does not have to manage addresses each time a user requests one. DHCP has some other interesting advantages. For example, if you have a limited number of IP addresses for a higher number of systems that are not always used at the same time, you can use DHCP to automatically assign addresses to systems that are actually running.

Another example is the use of mobile computers, such as laptops. These computers often go from one location to another location having different network parameters. DHCP allows dynamic assignment of an address to these temporary hosts very easily. As with any other TCP/IP protocol, DHCP is a client/server protocol. In AIX, the DHCP client daemon is called dhcpcd, and the DHCP server daemon is called dhcpsd.

10.2.11.3 Starting a DHCP client system

When you start a DCHP client for the first time, it searches for a DCHP server by using the 0.0.0.0 IP address and sends a DHCP discover packet (UDP packet).

All the DHCP servers defined in the local network (the broadcast does not cross IP routers) will check if they have an IP address available for that client, offer that address plus associated network information, and temporarily reserve the address. The DHCP client responds by broadcasting a DHCP request packet after selecting the first address it received. The other DHCP servers that reserved an IP address which has not been selected, unreserved that address.

The DHCP server whose offered address has been selected sends an acknowledgment by broadcast to the DHCP client. The client then has all the information it needs to fully work on the TCP/IP network. The network information is cached on the local client system and can be reused after a reboot.

If the DHCP server is on a different network, in order for the broadcast to cross the router, the router must support the BOOTP/DHCP relay (RFC 1542).

10.2.11.4 DHCP automatic allocation

DHCP can assign a permanent address to the client host.

10.2.11.5 DHCP dynamic allocation

DHCP can assign an address for a limited period of time. This kind of address is called a lease. Associated with the lease is a lease period. After the expiration of that period, the lease is returned to the pool of addresses to which it belongs.

The client must renew its lease after 50 percent of the lease time has been used. The network administrator can define a pool of addresses called scopes. A lease period can be determined for each scope.

10.2.11.6 DHCP manual allocation

The network administrator sets a specific IP address to a specific host known by its hardware address.

10.2.11.7 DHCP on AIX features

In AIX, the DHCP server allows the network administrator to do the following:

- Create network and subnet addresses.
- Create scopes (or pools) of addresses for each network and/or each subnet.
- Modify pools of addresses by adding or removing addresses in the pool.
- Exclude addresses from a pool of addresses.
- Reserve an address for a specific client by using its MAC address, its host name, or its client ID. This is a way of guaranteeing that a system will always get the same address. This mechanism is known as the client reservation.
- Client ID is a string of characters that allows the grouping of machines with similar network characteristics. For example, all machines sharing the same resources can be grouped by the same client ID. It is a way of grouping machines differently than the traditional network, subnet.
- Define a list of routers and their preferences.
- Define a list of DNS servers and their preferences.
- Reserve addresses to specific clients.

Besides, with AIX 4.3.3, some enhancements are added to the DHCP server program. They include:

- **Dynamic DNS update daemon** - the program that updates the DNS server now runs as a daemon and results in a performance improvement.
- **User defined extension** - The AIX 4.3.3 DHCP server provides a way to extend the DHCP server using user-defined extension. The extension,

also called DHCP server API, is provided primarily for integrating DHCP server with IP management software packages.

10.2.12 Dynamic Domain Name System (DDNS)

While DHCP dynamically assigns IP addresses, the DHCP client host name still needs to be manually registered by the network administrator to the domain system.

10.2.12.1 Design

The Domain Name System was originally designed to support queries of a statically-configured database. While the data was expected to change, the frequency of those changes was expected to be fairly low, and all updates were made as external edits to a zone's Master File.

The Dynamic Domain Name System (DDNS) protocol defines extensions to the Domain Name System to enable DNS servers to accept requests to update the DNS database dynamically.

10.2.12.2 Update operation

When a zone is modified by an UPDATE operation, the server must commit the change to non-volatile storage before sending a response to the requester or answering any queries or transfers for the modified zone.

10.2.12.3 Client point of view

From a requester's point of view, any authoritative server for the zone can appear to be able to process update requests, even though only the primary master server is able to modify the zone's master file. Requestors are expected to know the name of the zone they intend to update and to know (or be able to determine) the name servers for that zone.

DHCP can be used in conjunction with DDNS. When the DHCP server offers an IP address to a DHCP client, it can issue an inverse domain name query to a DDNS server to check if there is an existing mapping of a name to the IP address that the DHCP server is about to offer. If there is, the DHCP server will include this hostname in its offer, and the DDNS client can use it to update the DDNS server accordingly.

10.2.12.4 Limitations

It is not possible to create a zone using this protocol since there is no provision for a slave server to be told who its master servers are. It is expected that this protocol will be extended in the future to cover this case.

10.2.12.5 Security

The DNS server's integrity is critical to the smooth operation of the network. Therefore, IBM has provided security extensions to the DNS protocol as defined by the IETF DNS Security Working Group.The customer must decide to activate the security extensions. Once the security extensions are activated, RSA public-key digital signature technology is used to help secure the DNS updates.

10.2.13 Basic Network Utilities (BNU/UUCP)

AIX supports Basic Network Utilities (BNU/UUCP). These utilities have been very common in the UNIX community in the past. They provide tools and commands to connect to a remote system, copy files from one system to another, encode and decode binary data, queue jobs, execute commands on remote systems, and so on.

10.2.14 Mail

AIX includes the sendmail program and the Rand Corporation Message Handler (MH) application, which allow users to generate, process, send, and receive messages across a TCP/IP network.

10.2.14.1 Sendmail

The latest release of sendmail is ported to AIX 5L. On AIX 5L, anti-spam features are not activated by default; we recommend that you generate or modify the sendmail.cf file with anti-spam features.

What is spam?

E-mail spam is copies of the same message sent to people who do not want the message. Typically, it contains commercial advertisements or malicious messages. It wastes the CPU time and disk space of mail servers, the bandwidth of network lines, and readers' time and money. Sometimes, spammers use mail servers to relay their messages to numerous destination addresses. They send only one message to mail severs, and mail servers broadcast the message to the destination addresses listed in the RCPT TO: field. This method is called third party relay. In some cases, spammers hide their original e-mail addresses or original host addresses by changing some fields in their messages to non-existent ones or those of other organizations, especially when the messages are malicious.

Enhancements added by IBM

The following enhancements are added by IBM against sendmail 8.9.3.

- **Name Resolution Order** - Sendmail uses the AIX host resolution ordering mechanism. First, the NSORDER environment variable is queried. If it is not defined, sendmail searches /etc/netsvc.conf and then /etc/irs.conf. Sendmail's /etc/services.switch file is searched last. If none of them are found or the host entry is not defined, the system-wide default ordering is used.

- **IP v6 Support** - IP v6 is supported as an underlying protocol.

10.2.14.2 POP3 and IMAP4 support

AIX provides support for Post Office Protocol Version 3 (POP3) and Internet Message Access Protocol Version 4 (IMAP4). POP3 conforms to RFC 1725, and IMAP4 conforms to RFC 1730.

The POP server performs Message Transfer Agent (MTA) for a disconnected end user. The POP server holds messages, and, when the POP client (mail user agent) connects and requests messages, the POP server downloads all pending messages to the client machine. Most of the mail clients today operate using POP.

IMAP server performs online and disconnected access.

10.2.15 X11 protocol support

The following sections deal with X11 protocol support.

10.2.15.1 Xstation manager

AIX supports multiple concurrently logged users. This allows several Xstation users to simultaneously work on the same system. AIX provides a program called Xstation Manager to manage multiple Xstation connections.

10.2.15.2 X Display Manager Support (XDM)

The X Display Manager, XDM, runs as a daemon on a host machine. It provides a way for users to log on and start initial clients, regardless of what X Server they use. The XDM program is simply an X client that manages the first and last points of connection, control, and coordination of the user session. XDM incorporates the X session startup process into the login process. When set up properly, XDM enables users to walk up to a display and log in by typing their usernames and passwords just as they would on an ASCII terminal. XDM then runs their setup scripts automatically, setting up customized environments and enabling users to begin work immediately. When users finish their X sessions, XDM resets each display for the next user.

10.2.15.3 Common Desktop Environment Support (CDE)

AIX CDE, the graphical user interface to AIX Version 4, provides an easy-to-use, intuitive way to manage the user environment. AIX CDE is the default desktop for AIX. CDE is X/Open branded. The CDE login manager is responsible for providing a user login facility that authenticates and then initiates a desktop session for a given user.

The Login Manager manages a collection of X displays, both local and remote.

The emergence of X terminals guided the design of several parts of this system along with the development of the X Consortium standard XDMCP (X Display Manager Control Protocol). The Login Manager provides services similar to those provided by init, getty, and login on character terminals.This includes prompting for login ID and password, authenticating the user, and starting a session.

10.2.15.4 X11R6.3

X11R6.3, known as Broadway, is ported from X consortium to AIX 5L. X11R6.3 is upward compatible to X11R6, and X11R6.1. X11R6/6.1 clients can communicate with X11R6.3 servers. The major new functionality in X11R6.3 is support for the World Wide Web. It addresses secure and browser-oriented communication between X servers and clients through a low bandwidth network. Broadway allows X applications to be distributed via Web protocols, that is, a user may browse a corporate intranet (or the Internet) and click on a link within an HTML page to launch an X client. The application will then display within the Web-browser. X11R6.3 contains the following new functionalities:

- **Low Bandwidth X (LBX)** - Low Bandwidth X, formerly known as X.fast, is an extension to X server. It accelerates interactions between X clients and servers by using compression and caching techniques. By utilizing LBX, you can run X applications on a remote server that are displayed on your local X server at moderate response time even if the remote server is connected through WAN or dial-up modems.
- **Remote Execution (RX)** - The remote execution (RX) service of X11R6.3 defines a MIME format for invoking X applications remotely. By specifying application/x-rx as MIME type, an application, such as Web browser, can display X application on its window or local X server typically using plug-ins or helper applications.
- **Universal access** - UA is provided by integrating X11R6.3 with HTTP protocols making it platform-independent, thus, widely available.

- **Security extension** - X11R6.3 provides security extension that contains new protocols to enhance X server security. Clients issue a SecurityGenerateAuthorization request to X servers, and, if the request is successfully accepted by the servers, the clients are trusted; otherwise, they are untrusted. Untrusted clients have limited funtionalities and some restrictions.
- **Application group extension** - By using application group extension, applications other than window managers can manage other applications' windows. The primary purpose of this extension is for applications, such as Web browsers, to embed or insert other applications' windows into their windows.

Who needs Broadway?

Anyone who wishes to deliver access to corporate applications via an intranet or the Internet is a candidate for X WEB. The alternative method available today for launching graphical UNIX applications via a Web browser include Java. The Java method involves recoding applications and employing less than acceptable graphics widgets. The investment protection of X Web makes it an ideal solution for enterprise deployment via Web protocols. Java requires a rewrite of existing code to Web-enable applications. X11R6.3 (BROADWAY) was designed to be transparent to applications and requires no recoding. Both Java and X11R6.3 (BROADWAY) have a future in the Web-delivery of applications. X11R6.3, however, has the advantage of protecting existing technology investments.

10.2.16 Network Time Protocol (NTP)

NTP provides a mechanism for synchronizing time and coordinating time distribution in large Internet networks. The xntpd daemon is a complete implementation of the Network Time Protocol (NTP) Version 3 standard, as defined by RFC 1305, and also retains compatibility with Version 1 and 2 servers as defined by RFC 1059 and RFC 1119, respectively.

10.2.17 License management

The License Use Management function is provided by the License Use Manager Runtime. It consists of the server and client components required for controlling the license-enabled products that operate in a network. Any number of license-enabled products can be supported once the environment has been set up and configured. More than one license server can operate simultaneously in a network. The license passwords are entered into one or more license servers depending on the configuration. When a license-enabled product (a license client) requests a license, one of the license servers containing a license satisfies the request.

10.2.18 Internet/intranet software

This section describes some Internet/Intranet software available on AIX.

10.2.18.1 Netscape browser

AIX 5L bonus pack includes Netscape Communicator 4.75 with 128-bit encryption. The Web browser allows Web pages to be accessed from the Internet or intranet and viewed locally on the client desktop. It also allows access to e-mail and News. The built-in scripting language, called JavaScript, is supported in Netscape Communicator. JavaScript extends and enhances the capabilities of HTML documents. Java applets are also supported.

Netscape communicator Version 4.7.5.0 includes the following features:

- Navigator, Messenger, Newsgroups, and Composer
- Message filesets for Brazilian, Protugese, Catalan, Czech, English, French, German, Hungarian, Italian, Japanese, Korean, Polish, Russian, Slovakian, Spanish, Simplified Chinese, and Traditional Chinese.
- Help fileset for English
- Initial unicode (UTF-8) implementation for all languages listed above for AIX 5L.
- Bi-directional support for Hebrew and Arabic locales
- IBM's techexplorer Hypermedia Browser 3.0 Preview Release 1, a Web browser plug-in, for people who read or publish scientific articles, books and journals in the popular LaTeX, TeX, and the World Wide Web Consortium XML-based Mathematical Markup Language. For more information about techexplorer, go to the following URL:

  `http://www.software.ibm.com/software/techexplorer`
- Java support, JVM 1.1.5 and AWT 1.1, used by the AIX Web-based System Manager in applet mode.

10.2.18.2 Java

AIX 5L CD includes IBM's implementation for AIX of Sun's Java programming environment, based on Sun's Java Developer's Kit, Java 2 Technology Edition, Version 1.3 (Java 1.3.0).

Java 2 Standard Edition Version 1.3 comes with a host of enhancements to Java classes and APIs, including the user interface, graphics, sound, networking and math libraries. Of course, IBM's implementations, which are fully compliant with J2SE 1.3, have these enhancements too. IBM AIX Developer Kit, Java 2 Technology Edition, Version 1.3 has been engineered to with the following features to deliver high performance and scalability to the most demanding e-business applications:

- The latest version of Just-In-Time compiler version 3.6 with mixed mode interpreter for selective compilation of frequently executed code
- Efficient management of large Java heaps through optimized object allocation and efficient garbage collection
- Efficient and granular lock implementation
- Efficient exploitation of native AIX threads
- Robust networking supporting a large number of concurrent connections

Other highlights of IBM AIX Developer Kit, Java 2 Technology Edition, Version 1.3 are:

- Remote Method Invocation-Internet Inter-Object Request Broker Protocol (RMI-IIOP) extends the base JAVA RMI to perform communication using the Common Object Broker Architecture (CORBA) standard internet Inter-ORB Protocol (IIOP). For more information, see:

  `http://www.ibm.com/java/jdk/rmi-iiop/index.html`

- Java Naming and Directory Interface (JNDI) provides a unified interface to enterprise directory services such as CORBA's Common Object Services Naming Service, the Java RMI Registry and Lightweight Directory Access Protocol (LDAP).
- Security: Java Authentication and Authroization Service (JAAS), a major security component, provides a security model for the Java platform, which permits access to Java controlled resources based on the identity of the user on whose behalf the Java program is running, rather than the source of the code.
- Re-engineered Java VM from IBM: One of the major aspects of the re-engineering is a separation of the code base into components, with clearly defined interfaces between them. With the re-engineering of the JVM, it is possible to add significant serviceability aids to the core of Java technology.
- JDBC/ODBC Bridge allows access to databases with ODBC drivers.
- Java Communications API version 2.0 allows Java applications to access RS232 serial ports and IEEE 1284 parallel ports. For more information, see:

  `http://www.javasoft.com/products/javacomm/`

- Java Plug-In allows applets running in AIX's Netscape Communicator 4.7x Web browser to run using AIX Java version 1.3.0.

- IBM Big Decimal Extension adds decimal floating point extension to Java's BigDecimal class. For more information see:

  ```
  http://www.ibm.com/java/jdk/decimal/
  ```

Additional information about IBM AIX Developer Kit, Java 2 Technology Edition, Version 1.3 is available at http://www.developer.ibm.com/java/j2/index.html. IBM AIX Developer Kit, Java 2 Technology Edition is also available on the Web:

```
http://www.ibm.com/java/jdk/aix.
```

10.2.18.3 IBM HTTP Server Version 1.3.13 (powered by Apache)

IBM has enhanced the Apache HTTP Server with performance and SSL for secure transactions. And when serving static content, the HTTP Server may also see up to 40% performance improvement when used with in-kernel HTTP Get Engine in AIX 5L.

10.2.19 AIX Network Programming Interfaces

This section highlights some network APIs available in AIX.

10.2.19.1 AIX streams

STREAMS is a framework for the implementation of byte-oriented (vs. block-oriented) communications services.

- This framework consists of kernel resources and routines that can be used for:
  - Device driver creation
  - Terminal handlers
  - Networking protocol suites
  - Other networking facilities
- STREAMS was designed to unify the disparate (and often ad-hoc) mechanisms that previously existed in the UNIX operating system to support various types of character-based I/O. In particular, it was intended to replace the clist mechanism that provided support for terminal I/O.
- AIX Version 4 STREAMS is based on the OSF/1.2 Portable Streams Environment (PSE).
- STREAMS allows users to add (push) and remove (pop) intermediate processing elements, called modules, to and from the data stream at will.

- Because functions can be coded into separate STREAMS modules, developers can write kernel and applications programs that are highly-portable and easily integrated into other STREAMS-based systems.
- The major components of AIX (OSF/1) STREAMS are:
  - Data structures, declared constants, macros, and other kernel resources that developers use for constructing STREAMS modules and drivers.
  - The stream head, a set of functions and data structures that provide an interface between user processes and the streams constituting communications paths. In AIX Version 4 MP environments, the stream head also contains a special set of data structures and synchronization functions that enable streams to operate in a multi-threaded environment.
  - Utilities that perform functions, such as stream queue scheduling and flow control, memory allocation, and callback requests.
- AIX Version 4 STREAMS (OSF/1) is source-code compatible with the AT&T System V Release 4.

10.2.19.2 AIX sockets

AIX is based on BSD 4.3 Reno and BSD 4.4 software. The Berkeley socket interface provides generalized functions that support network communication using many possible protocols. Socket calls refer to all TCP/IP protocols as a single protocol family. The interface allows the programmer to specify the type of service required rather than the name of a specific protocol.

10.2.19.3 AIX pipes

Pipes are implemented as streams in AIX Version 4. There are two kinds of pipes:

- **Unnamed pipe** - An unnamed pipe (also called an “anonymous pipe”) is so called because it has no entry in the file system name space.
- **Named pipe** - A named pipe is also called a FIFO because data are received in a first-in-first-out manner. A named pipe is created via the mknod() system call. It has a name in the file system and can be accessed with the open() system call.

10.2.20 Distributed Computing Environment (DCE)

This section gives an overview of OSF DCE.

10.2.20.1 OSF DCE

The OSF Distributed Computing Environment (DCE) is a set of integrated services designed to support the development and use of distributed applications. OSF DCE is operating system-independent and provides a solution to the problem of sharing resources in a heterogeneous networked environment. This is accomplished by providing the services necessary to create an environment where a group of networked machines can share and manage resources. This allows for efficient use of present technology and for new technology to be incorporated as it becomes available.

DCE provides interoperability and portability across diverse operating systems and hardware platforms. DCE is a complete framework on which you can develop and maintain your distributed applications and services in an environment that you can scale to address the changing requirements of your enterprise.

10.2.20.2 DCE architecture

DCE is a layer of services that allows distributed applications to communicate with a collection of computers, operating systems, and networks. This collection of machines, operating systems, and networks, when managed by a single set of DCE services, is referred to as a DCE cell. Figure 137 illustrates the DCE architecture by showing the different components of DCE.

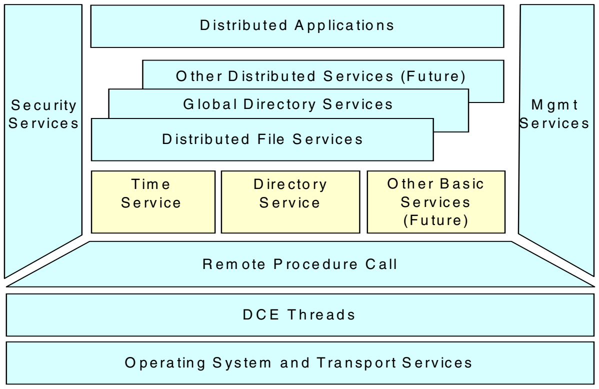

Distributed Applications
Other Distributed Services (Future)
Global Directory Services
Distributed File Services
Security Services
Mgmt Services
Time Service
Directory Service
Other Basic Services (Future)
Remote Procedure Call
DCE Threads
Operating System and Transport Services

Figure 137. DCE architecture

DCE components use three distributed computing models:

- **client/server model** - In this model, a distributed application is divided into two parts: The client and the server. In simple terms, the client is the entity that initiates the request for a service. The server is the entity that handles the request for a service. Note that the client and the server part can run on the same machine or on different machines.

- **Remote Procedure Call model (RPC)** - In this model, the client calls a remote procedure that appears as local. The procedure call is translated, and network communications are handled by the RPC mechanism. The server receives a request and executes the procedure, returning the results to the client. DCE RPC is an implementation of this model and is used by most of the other DCE technology components for their network communications.
- **Data Sharing model** - While client/server and RPC are focused on distributed execution, data sharing is concerned with how data is distributed among several clients and servers. Data sharing must address such needs as multiple copies of data, data consistency, and managing simultaneous access to data. In DCE, data sharing is built upon the RPCs, which are used as the means of transferring data.

10.2.20.3 DCE Cell

A collection of machines that are managed together as a DCE unit are referred to as a cell in DCE terminology. At a minimum, a cell must contain a Security Server and a Cell Directory Server. All of these services may run on one machine, or the servers can be spread among the machines that are to be part of the cell. The Security, Directory, and Time Services are collectively known as the DCE core services.

10.2.20.4 DCE security service

Most multiuser operating systems provide some method of verifying the identity of a user (authentication) and determining whether a user should be granted access to a resource (authorization). In a distributed environment, a way has to be provided to authenticate requests made across the network and to authorize access to the network's resources. There must also be a mechanism to protect network communications from attack. The challenge in a distributed environment is to provide these services transparently to both users and programs. For example, a user should not have to authenticate each server in the network. The DCE Security Service can provide this level of functionality because of how it has been integrated with the other DCE services.

The DCE security specification was submitted by MIT. It is based on Kerberos Version 5.1. Kerberos is an authentication service that validates the identity of a user or service. The DCE Security Service is made up of several parts.

- The Authentication Service that allows processes on different machines to determine each other's identity (authenticate).

- The Privilege Service that determines if an authenticated user is authorized to access a server resource. The Privilege Service provides information that servers need to determine the access that should be granted to the user.
- The Registry Service that manages a security database used to hold entries for all principals. A principal is a user, an application server, or a computer.
- The Registry Service is also used by administrators to maintain the list of principals known to DCE.
- The Audit Service that detects and records security operations performed by DCE servers.
- The Login Facility that performs the initialization of the DCE environment for a user. It uses the Security Service to authenticate a user and returns credentials to the user. These credentials are then used to authenticate other services in the DCE cell. The credentials expire after a set period of time and need to be refreshed when they are needed beyond the preset time.

10.2.20.5 DCE Directory Service

The Directory Service provides a naming model that allows users to identify, by name, network resources, such as servers, users, files, disks, or print queues.

The DCE Directory Service includes:

- Cell Directory Service (CDS)
- Global Directory Service (GDS)
- Global Directory Agent (GDA)
- Application Programming Interface (API)

The CDS manages information within a cell.The GDS is based on the CCITT X.500 name schema and provides the basis for a global name space. The GDA is the CDS gateway to intercell communication. The GDA supports both Internet addresses and X.500 addresses. If the address passed to the GDA is an X.500 address, the GDA contacts the GDS. If the address passed to the GDA is an Internet address, the GDA uses the Internet Domain Name System (DNS) to locate the foreign cell. Both CDS and GDS use the X/Open Directory Service (XDS) API as a programming interface.

10.2.20.6 DCE Distributed Time Service

Distributed Time Service (DTS) provides precise, fault-tolerant clock synchronization for the computers participating in a Distributed Computing Environment, both over LANs and WANs. The synchronized clocks enable DCE applications to determine event sequencing, duration and scheduling. DTS is based on Universal Coordinated Time (UTC) time, an international time standard.

10.2.20.7 Distributed File System (DFS)

The Distributed File System (DFS) presents directories and files in a global name space that can be accessed from any DFS client. Caching on DFS clients reduces access time and network traffic and results in high performance.

DFS includes support for both Journaled File System (JFS) and Local File System (LFS) formats. LFS is a fast-restarting, log-based physical file system that supports file replication for high availability.

DFS files and directories can be protected by using Access Control Lists (ACL). You can define an ACL for each file or directory to restrict or authorize access. With DFS ACLs, you have the granularity to control access for users and groups of the local or any foreign cell. DFS ACLs are different than the access control list support provided through the AIX operating system.

The DFS is built on top of the core technologies: Security Service, Cell Directory Service, and Distributed Time Service. DFS also makes use of threads and DCE RPCs. It implements a superset of the POSIX 1003.1 file system semantic standard submitted by Transarc Corporation and is based on the Andrew File System. In addition, DFS:

- Builds on the fundamental features of DCE: Security, threads, RPC, directory, and time services
- Allows transparent access to files anywhere on the network, including AIX CD-ROM file systems
- Improves administration through file sets
- Boosts reliability through a log-based physical file system and through the data replication feature
- Provides authenticated access from any Network File System (NFS) client through the DCE NFS to DFS Authenticating Gateway

10.2.20.8 Threads

Threads support the creation, management, and synchronization of multiple paths of control within a single process. The threads programming facility is also POSIX 1003.4a Draft 4 compliant. If threads are already available on the operating system, DCE can use them.

10.2.20.9 Remote Procedure Call (RPC)

This is a complete environment to help develop client/server applications. A development tool, consisting of an Interface Definition Language (IDL), is provided. The RPC runtime service facilitates the implementation of the network protocols used by the client and server applications to communicate. One component of the RPC is the uuidgen, a program that generates a Universally Unique Identifier (UUID, a 32-digit number) to uniquely identify resources, services, and users in DCE independently of time and space. The RPC specification is based on Network Computing System architecture (NCS) Release 2.

10.2.20.10 IBM added value

IBM has added several components to the base OSF DCE offering on AIX to provide the following functions:

- DFS access from NFS clients with the NFS to DFS Authentication Gateway for AIX

  The NFS to DFS Authentication Gateway is a product on the AIX platform that allows NFS client systems to access the DFS file space. The NFS client is typically run on a system that is not part of the DCE cell or one that has no DFS code installed.

  The gateway is installed on a DFS client system that exports its DFS file system into NFS, thus, acting as an NFS server. The gateway provides a bridge between the authentication methods of DFS and NFS. This is accomplished by connecting an NFS client with a DCE principal. The NFS to DFS Authentication Gateway allows the NFS client to obtain authenticated access to the DFS file space.

- **Exportable data encryption** - The RPC communication provides different security levels, the highest being full data encryption. However, the DES algorithm (Data Encryption Standard) internally used by DCE cannot be exported outside the U.S. in a user accessible form. This means it cannot be used for data encryption.

  On the AIX platform, there is a User Data Masking Facility, which is still referred to as a Common Data Masking Facility or CDMF. CDMF allows you to encrypt user data in RPCs using DES with a 40-bit key instead of the standard 56-bit key. Since this makes the encryption weaker, it has no export restrictions from the U.S. It is a good solution for non-U.S. customers who want increased privacy but cannot have an export license for full DES.

- **Online documentation** - All DCE manuals are provided in softcopy form to be accessed with a graphical viewer.

  The IBM DCE Version 2.1 for AIX provides an IPF viewer for X Window (IPF/X). The `xview` command that starts IPF/X provides hypertext linking, search, and print facilities, inline graphics display, a bookmark function, and online help. Its startup is integrated into InfoExplorer. IBM DCE 2.1 for AIX also provides the documentation in ASCII that can be viewed from ASCII terminals with an ASCII browser. The `dceman` command emulates MAN pages for DCE commands.

- **Update procedure** - DCE for AIX uses the same update procedures as AIX with the support of the `installp` command. Version - Release - Modification - Fix (VRMF) is the strategy for the fixes applied to the system. System maintenance and release level verification is extremely powerful and easy to use for the system administrator.

10.2.20.11 DCE enablement of applications

DCE is a framework and a basis for distributed computing; the enablement of applications is a very important factor for the success of this great piece of technology. IBM is currently looking at how to integrate DCE with other products:

- **DCE integration with ADSM** - The ADSM client and server software will be enhanced to provide integration with DCE/DFS. ADSM will be able to backup and restore DFS files and access list control information. In addition to that, the DFS fileset backup facility can use ADSM as a backend storage provider.
- **DCE integration with AIX Fast Connect** - AIX Fast Connect Release 3.1 for Windows and OS/2 includes integration with DCE/DFS. This feature allows user authentication with the DCE security server. DFS directories can be shared with (exported to) PC clients. PC client access is controlled by the login context acquired as a result of DCE authentication. Fast Connect offers DCE authentication and DFS access without requiring DCE/DFS client software to be installed on each of the PC clients requiring DCE/DFS access.
- **DCE Event Management Service (EMS)** - The DCE Event Management Service (EMS) supports asynchronous event management for use by system-management applications. EMS uses the concepts of event suppliers and event consumers and sets up an event channel between

them to support asynchronous communication. In the context of DCE, event suppliers are any DCE core service or DCE-based application (client or server), and event consumers can be any application with an interest in receiving asynchronous events from one or more DCE processes. The transmission of events between suppliers and consumers is uncoupled by routing events via EMS, which is the implementation of an event channel. EMS also provides a filtering mechanism to allow administrators and consumers control over which events EMS will send. EMS provides integration for DCE clients and servers using the DCE Serviceability (SVC) interface. DCE applications can use the APIs offered in SVC to become event suppliers.

10.2.21 Remote access

This section describes two main protocols supported on AIX for accessing an AIX system remotely.

10.2.21.1 Point-to-Point Protocol (PPP)

The Point-to-Point Protocol (PPP) is an Internet Engineering Task Force (IETF)-defined protocol for both synchronous and asynchronous connectivity. It has become popular for asynchronous TCP/IP access of servers. This connectivity is very popular for applications, such as those used for Internet access. The AIX PPP in AIX implementation adheres to IETF Request for Comments (RFCs) 1661, 1332, and 1662. It is a streams-based kernel implementation, in contrast to the daemon implementation of some commercially available packages. The kernel implementation prevents the performance overhead of having part of the protocol in application space. It was designed to provide a robust TCP/IP server attachment.

The Point-to-Point Protocol (PPP) has been enhanced to include PAP (Password Authentication Protocols) and CHAP (Challenge Handshake Authentication Protocol) to provide authentication of remote users. It conforms to the RFC 1334 PPP Authentication.

10.2.21.2 Serial Line Internet Protocol (SLIP)

SLIP has its origins in the 3COM UNET TCP/IP implementation from the early 1980s. It is merely a packet-framing protocol: SLIP defines a sequence of characters that frame IP packets on a serial line, and nothing more. It provides no addressing, packet-type identification, error detection/correction, or compression mechanisms. However, because the protocol does so little, it is usually very easy to implement. SLIP is part of AIX and adheres to the RFC 1055.

10.2.22 X.25 for AIX

This section describes the X.25 support on AIX.

10.2.22.1 General

X.25 networks have the ability to store and forward data. That is, they can receive an entire sequence of data and then route it to its destination based upon usage of the possible routes. Due to the nature of the X.25 protocol, it is possible that two individual data packets may take different routes through the same network and still arrive in the correct sequence. While there are many possible reasons for this, it is important to understand that X.25 preserves the data transmission order at the destination. X.25 virtual circuits allow the user to have multiple applications that communicate independently running between multiple machines.

10.2.22.2 Supported standards

The X.25 software supports the 1980, 1984, and 1988 CCITT standards.

10.2.22.3 X.25 software features

The X.25 software on AIX offers the following features:

- Three programming interfaces (NPI, DLPI, and COMIO)
- TCP/IP and SNA support
- Triple-X PAD including PAD printing
- SNMP support
- Increased throughput
- Power Management support on PCI/ISA bus systems

10.2.23 ISDN

This section describes Integrated Services Digital Networks (ISDN) support on AIX.

10.2.23.1 General

ISDN stands for “Integrated Services Digital Networks”, and it is an ITU-T (formerly CCITT) term for a relatively new telecommunications service package. ISDN is, basically, the telephone network turned all-digital, end-to-end, and using existing switches and wiring (for the most part) upgraded so that the basic “call” is a 64 Kbps end-to-end channel, with bit-diddling as needed (but not when not needed!).

Packet, and maybe frame modes, are also thrown in for good measure in some places. It is offered by local telephone companies, but most readily in

Australia, Western Europe, Japan, Singapore, and some portions of the USA. In France, ISDN is known as RNIS.

10.2.23.2 Basic Rate Interface

A Basic Rate Interface (BRI) is two 64 Kb bearer (“B”) channels and a single delta (“D”) channel. The B channels are used for voice or data, and the D channel is used for signaling and/or X.25 packet networking. This is the variety most likely to be found in residential service.

Equipment known as a Terminal Adapter (TA) can be used to adapt these channels to existing terminal equipment standards, such as RS-232 and V.35.

Typically, this equipment is packaged in a similar fashion to modems, either as stand-alone units or as interface cards that plug into a computer or various kinds of communications equipment, such as routers or PBXs. TAs do not interoperate with the modem; they replace the modem.

10.2.23.3 Primary Rate Interface

Another flavor of ISDN is the Primary Rate Interface (PRI). In North America and Japan, this consists of 24 channels, usually divided into 23 B channels and 1 D channel, and runs over the same physical interface as T1. Outside these areas, the PRI has 31 user channels, usually divided into 30 B channels and 1 D channel and is based on the E1 interface. It is typically used for connections, such as one between a PBX (private branch exchange - a telephone exchange operated by the customer of a telephone company) and a CO (central office of the telephone company) or IXC (inter-exchange carrier - a long distance telephone company).

10.2.23.4 Data encapsulation for IP over ISDN

A decision was made at the Amsterdam IETF to state that all systems wishing to guarantee IP interoperability should implement PPP. Such systems may also implement the Frame Relay or X.25 encapsulations. An RFC will be published delineating how the encapsulations are limited to that set of three. They may be distinguished by an examination of the first correctly checksummed and HDLC bit-stuffed packet.

Many implementations are using PPP so that they can negotiate compression and/or multilink operation. A common practice in most European countries is to use raw IP packets delimited by HDLC flags. Another common practice is an encapsulation using simple HDLC in Layer 1, X.75 (LAPB, usually I-frames) in Layer 2, and, sometimes, T.70 in Layer 3. PPP is used instead of HDLC/X.75/T.70 when the network does not provide the caller’s telephone number (for example, when emulating a modem or the caller’s number is lost on telephone company borders). In this case, caller authentication is done via PAP/CHAP instead.

10.2.23.5 AIX ISDN Device Driver via netISDN

netISDN, from the German company NetCS, is a high-end ISDN solution for AIX. IBM resells the netISDN and netGW software for AIX. For more information, point your Web browser to:

```
http://www.netcs.com
```

10.2.24 ATM for AIX

Asynchronous Transfer Mode (ATM) is a switching/transmission technique in which data is transmitted in small cells of fixed-size (5-byte header, 48-byte payload). The cells lend themselves both to the time-division-multiplexing characteristics of the transmission media and to the packet switching characteristics desired of data networks. At each switching node, the ATM header identifies a *virtual path* or *virtual circuit*, for which the cell contains data, enabling the switch to forward the cell to the correct next-hop trunk. The virtual path is set up through the involved switches when two endpoints wish to communicate. This type of switching can be implemented in hardware, which is almost essential when trunk speed ranges from 45 Mbps to 1 Gbps.

There is a new feature with AIX 4.3.2 that helps increase performance when using the PCI ATM 155 adapter. The PCI ATM 155 adapter has a hardware TCP/UDP checksum capability. In AIX 4.3.2, the ATM network device driver is modified to use this feature. In the new ATM 155 adapter device driver, the workload of TCP data checksum processing is off-loaded from the AIX TCP/IP protocol stack to the adapter itself. A related enhancement automatically remembers the mapping of virtual addresses to physical addresses for the entire networking buffer pool to save address translation during networking I/O operations. TCP checksum offload for ATM reduces the amount of time spent on the main CPU computing checksums, thereby, reducing latency and allowing the system to handle more work, in particular, to handle more packets in the same amount of time. This results in performance improvements.

10.2.24.1 Multi-protocol Over ATM

Asynchronous Transfer Mode (ATM) networks provide a high-bandwidth network infrastructure. However, to help protect the investment of existing Local Area Network (LAN) clients, such as those that use traditional Ethernet connections, ATM offers LAN Emulation, or LANE clients, so that an ATM adapter can provide a traditional network interface. AIX Version 4.3.3 provides improved management and performance of an ATM LAN Emulation network. Device-specific configurations are minimized with auto-discovery and device discovery protocol, while data paths are reduced from many hops between routers to a single hop between end clients. In AIX Version 4.3.3, Multi-Protocol Over ATM (MPOA) supports both Standard Ethernet and IEEE 802.3 Ethernet.

To understand what MPOA does for you, an example from the AIX documentation is reproduced in Figure 138, along with a related discussion.

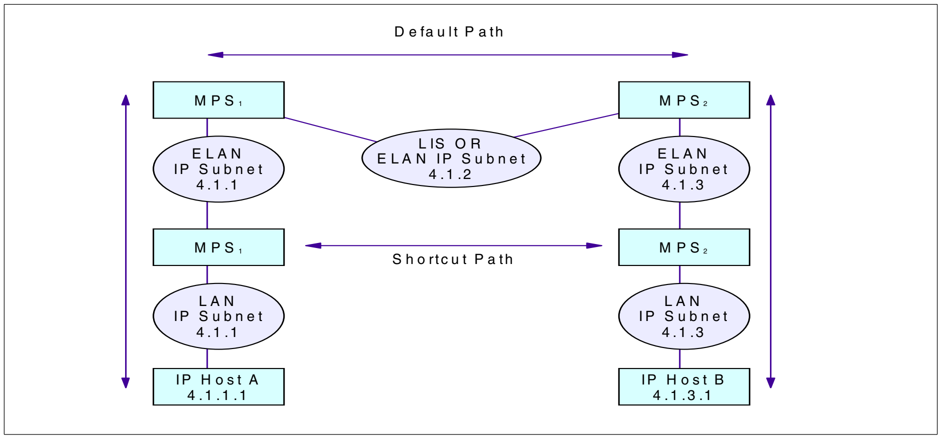


Figure 138. An example of a network that benefits from MPOA

In Figure 138 on page 422, the devices on the Emulated LAN (ELAN) network can be described as edge devices since they are an example of a device that can bridge packets between one or more LAN interfaces, such as Host A, and one or more LANE clients. The MPOA enhancement allows these edge devices to perform internetwork layer forwarding and establish direct communications without requiring that the LANE edge devices be full function routers.

So, using IP, packets from IP Host A would usually have to travel through the 4.1.2 subnet to be routed through to IP Host B. However, using MPOA, the 4.1.2 subnet can be bypassed. The MPC is the MPOA client that performs internetwork layer forwarding. The MPC obtains its forwarding information from the MPOA Server (MPS). The MPC detects the flow of packets being forwarded over an ELAN to a router that contains an MPS. When it recognizes a flow that could benefit from a shortcut, such as from Host A to B, it uses a Next Hop Resolution Protocol (NHRP)-based query-response protocol to request the information required to establish a shortcut to the destination. If a shortcut is available, the MPC caches the information, sets up a shortcut VCC, and forwards frames for the destination over the shortcut that bypasses the router. Note that IP is the only protocol stack supported. That is, SNA, IPX, and Netbios directly over ATM are not supported.

10.2.25 System Network Architecture (SNA) for AIX

The following section describes how AIX supports SNA.

10.2.25.1 General

In the early 1970s, IBM discovered that large customers were reluctant to trust unreliable communications networks to properly automate important transactions. In response, IBM developed Systems Network Architecture (SNA).

As the saying goes: Anything that can go wrong, will go wrong, and SNA may be unique in trying to identify literally everything that could possibly go wrong in order to specify the proper response. Certain types of expected errors, such as a phone line or modem failure, are handled automatically. Other errors, such as software problems, configuration tables, and so forth, are isolated, logged, and reported to the central technical staff for analysis and response. This SNA design worked well as long as communications equipment was formally installed by a professional staff. It became less useful in environments when any PC simply plugs in and joins the LAN. Two forms of SNA developed: Subareas (SNA Classic) managed by mainframes and APPN (New SNA) based on networks of minicomputers.

In the original design of SNA, a network is built out of expensive, dedicated, switching minicomputers managed by a central mainframe. The dedicated minicomputers run a special system called NCP. No user programs run on these machines. Each NCP manages communications on behalf of all the terminals, workstations, and PCs connected to it. In a banking network, the NCP might manage all the terminals and machines in branch offices in a particular metropolitan area. Traffic is routed between the NCP machines and, eventually, into the central mainframe.

The mainframe runs an IBM product called VTAM, which controls the network. Although individual messages will flow from one NCP to another over a phone line, VTAM maintains a table of all the machines and phone links in the network. It selects the routes and the alternate paths that messages can take between different NCP nodes.

A subarea is the collection of terminals, workstations, and phone lines managed by an NCP. Generally, the NCP is responsible for managing ordinary traffic flow within the subarea, and VTAM manages the connections and links between subareas. Any subarea network must have a mainframe.

The rapid growth in minicomputers, workstations, and personal computers forced IBM to develop a second kind of SNA. Customers were building networks using AS/400 and RS/6000 computers that had no mainframe or VTAM to provide control. The new SNA is called APPN (Advanced Peer-to-Peer Networking).

APPN and subarea SNA have entirely different strategies for routing and network management. Their only common characteristic is support for applications or devices using the APPC (LU 6.2) protocol.

10.2.25.2 SNA vs. TCP/IP

It is difficult to understand something unless you have an alternative with which to compare it. SNA is not TCP/IP. This applies at every level in the design of the two network architectures. Whenever the SNA designers went right, the TCP/IP designers went left. Ironically, instead of the two network protocols being incompatible, they turn out to be complimentary. An organization running both SNA and TCP/IP can probably solve any type of communications problem.

An IP network routes individual packets of data. The network delivers each packed based on an address number that identifies the destination machine. TCP is responsible for reassembling the pieces after they have been received.

In the SNA network, a client and server cannot exchange messages unless they first establish a session. In a Subarea network, the VTAM program on the mainframe gets involved in creating every session. Furthermore, there are control blocks describing the session in the NCP to which the client talks and the NCP to which the server talks. Intermediate NCPs have no control blocks for the session. In APPN SNA, there are control blocks for the session in all of the intermediate nodes through which the message passes. Control blocks describe the session in the NCP to which the client talks.

The APPN architecture was designed originally for minicomputers. APPN has two kinds of nodes. An End Node (EN) contains client and server programs.

Data flows in or out of an End Node, but does not go through it. A Network Node (NN) also contains clients and servers, but it also provides routing and network management. When an End Node starts up, it connects to one Network Node that will provide its access to the rest of the network. It transmits a list of the LUNAMEs that the End Node contains to that NN. The NN ends up with a table of its own LUNAMEs and those of all the ENs that it manages. Most of APPN is the set of queries and replies that manage names, routes, and sessions. Like the rest of SNA, it is a fairly complicated and exhaustively documented body of code.

Obviously workstations cannot maintain a dynamic table that spans massive networks or long distances. The solution to this problem is to break the APPN network into smaller local units each with a Network ID (NETID). In common use, a NETID identifies a cluster of workstations that are close to each other (in a building, on a campus, or in the same city). The dynamic exchange of LUNAMEs does not occur between clusters with different NETIDs. Instead, traffic to a remote network is routed based on the NETID, and traffic within the local cluster is routed based on the LUNAME. The combination of NETID and LUNAME uniquely identifies any server in the system, but the same LUNAME may appear in different NETID groups associated with different local machines.

10.2.25.3 CPI-C

The native programming interface for modern SNA networks is the Common Programming Interface for Communications (CPI-C). This provides a common set of subroutines, services, and return codes for programs written in COBOL, C, or REXX. It is widely available in softcopy on CD-ROM.

10.2.25.4 IBM eNetwork Communication Server for AIX

The IBM eNetwork Communications Server for AIX, Version 5.0, responds to enterprises' needs for interconnecting diverse networks. With this product, workstation users and applications can communicate with other workstations and central computer applications regardless of the networking protocols used in each system. Communications Server provides a powerful, multi-protocol, full-function gateway to clients on SNA and TCP/IP networks, plus support for a broad range of industry-standard networking protocols.

The eNetwork Communications Server for AIX provides the following characteristics to support business needs, such as:

- Integrated TN3720E Server support
- Includes Host On-Demand V1, which provides easy 3270 SNA application access from any Java-enabled Web browser
- APPC over TCP/IP and Sockets over SNA network communications
- Advanced peer-to-peer networking (APPN) network node and end node support including high-performance routing (HPR) and dependent LU requester (DLUR)
- Rich set of APIs to develop applications for distributed computing including eNetwork Host Access Class Library (Host Access API)

- Broad range of LAN and WAN connectivity options
- Web-based 3270 access with Host On-Demand Version 1
- Motif Administration tool for improved configuration and administration

Due to the features described above, eNetwork Communication Server for AIX is a good solution for the customer who needs to:

- Connect to existing SNA networks
- Establish new SNA networks
- Integrate TCP/IP and SNA networks
- Connect over WAN using SDLC or X.25 or over LANs using IBM Token-Ring, Ethernet, FDDI, Frame Relay, ATM LAN Emulation, or channel protocols
- Take advantage of the peer-to-peer distributed networking capabilities of advanced peer-to-peer networking (APPN) in existing SNA networks

For additional information about the product, you may refer to the following URL:

`http://www.software.ibm.com/enetwork/commserver/`

10.2.25.5 APPC over TCP/IP

The IBM AnyNet family, based on Multiprotocol Transport Networking (MPTN) technology, an open industry standard architecture, is designed to allow any application to run over any network protocol. This means you can add applications designed to run over different protocols without changing applications or modifying hardware.

With AnyNet APPC over TCP/IP, you can extend advanced program-to-program communication (APPC) or Common Programming Interface for Communications (CPI-C) applications to TCP/IP users without adding a separate SNA network.This allows AIX, APPC, or CPI-C applications, such as CICS/6000 or DB2/6000, to communicate between a centralized computer and workstations or between workstations across a TCP/IP network, without changing the applications.

10.2.25.6 Sockets over SNA

With AnyNet Sockets over SNA, you can add Berkeley Software Distribution (BSD) Sockets applications to existing SNA networks without adding a separate TCP/IP network. This allows users of AIX platforms to access Sockets applications, such as File Transfer Protocol (FTP), SNMP, Lotus Notes, and the NetView program, across an SNA network.

10.2.25.7 TN3270E

IBM SNA Client Access provides access to SNA networks for a wide range of TCP/IP clients. It is the software solution best suited for allowing easy access to S/390 and AS/400 computers. Running in conjunction with Communications Server, SNA Client Access works as a TCP/IP Telnet server, providing SNA network access service to client applications running anywhere in the TCP/IP network.

10.2.25.8 Gateway

The SNA Gateway function of the eNetwork Communications Server allows many SNA clients to go through a single communication server to one or more centralized computers. It also allows clients to access, on the fly, a backup host that shares the workload and improves resource availability. SNA Gateway allows you to preset and manage sessions, automatically logging off unattended workstations to open up access for other users. You can also channel the gateway function to concentrate thousands of sessions into the centralized computer using only one physical connection.

10.2.25.9 Communications Server for AIX features

The following are the features of Communications Server for AIX:

- **Complete connectivity** - Whether you want to connect networks over a Wide Area Network (WAN) using SDLC or X.25 or over a Local Area Network (LAN) using token-ring, Ethernet, Fiber Distributed Data Interface (FDDI), or direct-attached Channel, Communications Server is the solution for you.

  You can also use eNetwork Communications Server to connect multiple physical units (PUs) across a single physical adapter for token-ring, Ethernet, X.25, SDLC, FDDI, and Channel. Support for multiple PUs extends the number of supported LUs per adapter port available for all link types.

- **3270 Emulation** - Included with eNetwork Communications Server for AIX is a license for the IBM 3270 Host Connection Program.

- **Simplified configuration** - The eNetwork Communications Server has a streamlined organization of configuration profiles that helps you drastically reduce configuration time. You can update the configuration database even when the eNetwork Communications Server is running, and the new configuration information does not become part of the database until you verify that it is correct.

- **High Availability** - When problems occur, you can find and fix them quickly using the new SNA format utility. This utility formats complex link traces into easy-to-understand, fully-deciphered output files. Trace facilities for SNA flows, events, and first-failure data capture are also provided to help you resolve problems quickly.

- **Systems management** - You can use the Xsna graphical interface tool to display and manage your SNA resources easily. Based on X Window and Motif, the Xsna tool provides a familiar look and feel. Using the Xsna tool, you can display link and session information, start and stop resources, and turn on traces – all with the click of a mouse button. System Management Interface Tool (SMIT) is also available for easy management in the AIX environment.
- **Power programming** - eNetwork Communications Server is not just a powerful stand-alone network server; its sophisticated programming interfaces make it an excellent platform for programming and application integration. Communications Server provides a number of application programming interfaces (APIs) ranging from platform to device level, including LUs 0, 1, 2, 3, and 6.2 (CPI-C and APPC), Generic SNA (for device-level programming), and SNA Management Services.
- **Tools** - eNetwork Communications Server includes an SNA interactive transaction program generator (SNAPI) that provides help for developing APPC and CPI-C transaction programs. You can use this tool to quickly develop programs that interact with existing programs on any remote system that supports LU 6.2 including AIX, CICS, IMS, OS/400, Communications Manager/2, and Communications Server for OS/2 Warp.
- **High performance** - eNetwork Communications Server takes advantage of the Symmetrical Multiprocessor (SMP) technology by exploiting the parallel processing capabilities of this new technology.

10.2.26 AIX PC connectivity with AIX Fast Connect

There are several products available for AIX to provide PC integration solutions. Some of these products are provided and supported by IBM, and others are not. One of the most popular non-IBM products that runs very well on AIX is SAMBA, a free product developed and supported by the Internet Community. If you require information about SAMBA, you may refer to the official SAMBA URL at:

`http://www.samba.org/`

For more information about SAMBA, AIX Fast Connect and other products for AIX and Windows interoperability, you may refer to the redbook *AIX 5L and Windows 2000: Solutions for Interoperability*, SG24-6225.

10.2.26.1 AIX Fast Connect Overview

Because AIX Fast Connect uses industry-standard Microsoft networking protocols, PC clients can access AIX files and printers using their native networking client software. PC users can use remote AIX file systems directly from their machines like local file system, and access AIX print queues like local printers. AIX Fast Connect provide these services by implementing the Server Message Block (SMB) networking protocol on top of NetBIOS over TCP/IP (RFC-1001/1002).

Features

Important features of AIX Fast Connect include:

AIX-application standard and advanced features, including

- Tight integration with AIX, using AIX features such as thread, kernel I/O, file system, and security
- Maintenance and administrating using Web-based System Manager, SMIT, and command line
- Streamlined configuration
- Trace and log capabilities
- SendFile API support
- DCE/DFS integration
- Support for JFS-ACLs
- HACMP support, using server name aliases

Advanced SMB/NetBIOS features, including:

- SMB-based file and print services
- Passthrough authentication to NT
- Resource browsing protocol (Network Neighborhood)
- Network logon support, including roaming user-profile
- WINS client and proxy, and NBNS-server
- Opportunistic locking (oplock)
- B-node support
- Guest logon support
- Share-level security support
- Message from server to client
- Mapping of AIX long filenames to DOS 8.3 filenames

- Unicode representation of share, user, file, and directory names
- Mapping of PC-client usernames to AIX usernames
- Multiplexed SMB-sessions (for Windows terminal server support)

10.2.26.2 AIX Fast Connect Version 3

In AIX 5L, AIX Fast Connect Version 3.1 provides the following new functionalities:

Locking enhancements

Some applications require shared files between server-based applications and PC client applications. The file server requires lock mechanisms to protect these files against multiple modifications at the same time. Because of this, Fast Connect implements UNIX locking, in addition to internal locking, to allow exclusions based on file locks taken by PC clients. AIX 5L implements the following lock enhancements;

- Opportunistic locks take exclusive lock on the file when the exclusive opportunistic lock is granted; the file will be unlocked when the opportunistic lock is broken.
- SMB share modes are implemented with UNIX lock consistent with the granted open mode and share mode.

Per share options

These options are encoded as bit fields with the sh_options parameter of each share definition. These options must be defined when the share is created with the `net share /add` command, or set through the SMIT file share panel.

Per-share options currently allowed by `net share /add` are shown in the Table 22.

Table 22. Per-share value options

Parameter	Values	Default	Description
sh_oplockfiles	(0,1)	1	If oplocks=1, enables opportunistic lock on this share
sh_searchcache	(0,1)	0	If searchcache=1, enables search caching on this share
sh_sendfile	(0,1)	0	If sendfile=1, enables sendfile API on this share
mode	(0,1)	1	Mode=1, enables read/write access mode=0, enables read only access

PC user name to AIX user name mapping

AIX Fast Connect on AIX 5L allows the server administrator to configure the mapping of PC user names to AIX user names. When enabled, AIX Fast Connect tries to map every incoming client user name to a server user name, and then uses that server user name for further user authentication and AIX credentials.

Windows Terminal Server support

In AIX 5L, Fast Connect allows multiple SMB sessions over one transport session. In previous releases, Fast Connect was limisted to one SMB session per transport connection.

Search caching

In AIX 5L, Fast Connect allows you to enable search caching. If enabled, all the cached structures will compare their time stamps to the original files to check for modifications periodically. This feature improves file searching significantly.

10.2.27 Network security

AIX is built to strongly support security when connected to a public network.

10.2.27.1 IP security

AIX comes standard with IPsec included.

The RFC 1826 AH and the RFC 1829 ESP can be used together or alone. The Keyed MD5 algorithm is available for RFC 1826 AH, and the DES CBC 4, DES CBC 8, and CDMF algorithms can be used with RFC 1829 ESP.

The new 96-bit HMAC AH format and the new combined ESP with Authentication format can be used. These formats currently exist as IETF drafts. The HMAC MD5 and HMAC SHA1 algorithms can be used with this new AH format, and the DES CBC 4, DES CBC 8, CDMF, and the DES CBC MD5 combination algorithms can be used with the new ESP format. Replay protection can also be used with either of these new formats.

IPsec supported standards

AIX supports the following security standards:

- RFC 1825 - Security Architecture for the Internet Protocol
- RFC 1826 - IP Authentication Header
- RFC 1827 - IP Encapsulating Security Payload (ESP)
- RFC 1828 - IP Authentication Using Keyed MD5

- RFC 1829 - The ESP DES-CBC Transform
- RFC 2104 - HMAC: Keyed-Hashing for Message Authentication
- RFC 2085 - HMAC-MD5 IP Authentication with Replay Prevention

10.2.27.2 SOCKS protocol Version 5 support

AIX provides support for Socks Libraries. AIX Socks API allows generic TCP/IP applications to connect to hosts through a generic TCP/IP proxy using SOCKS protocol Version 5.

10.2.27.3 Public-Key Cryptographic Standards (PKCS) Support

AIX 5L offers an implementation of the cryptographic API PKCS#11 version 2.01 on the POWER platform. PKCS#11 is a de facto industry standard for accessing cryptographic hardware devices. AIX 5L for POWER offers support for IBM 4758 model 2 cryptographic coprocessor under operating system PKCS#11 shared object. The AIX 5L PKCS#11 implementation is enhanced to utilize future IBM cryptographic hardware devices through the same shared library. Application which are available to utilize PKCS#11 on AIX 5L include the iPlanet server suite on AIX 5L. For additional information on PKCS11, refer to the RSA laboratories Web site at:

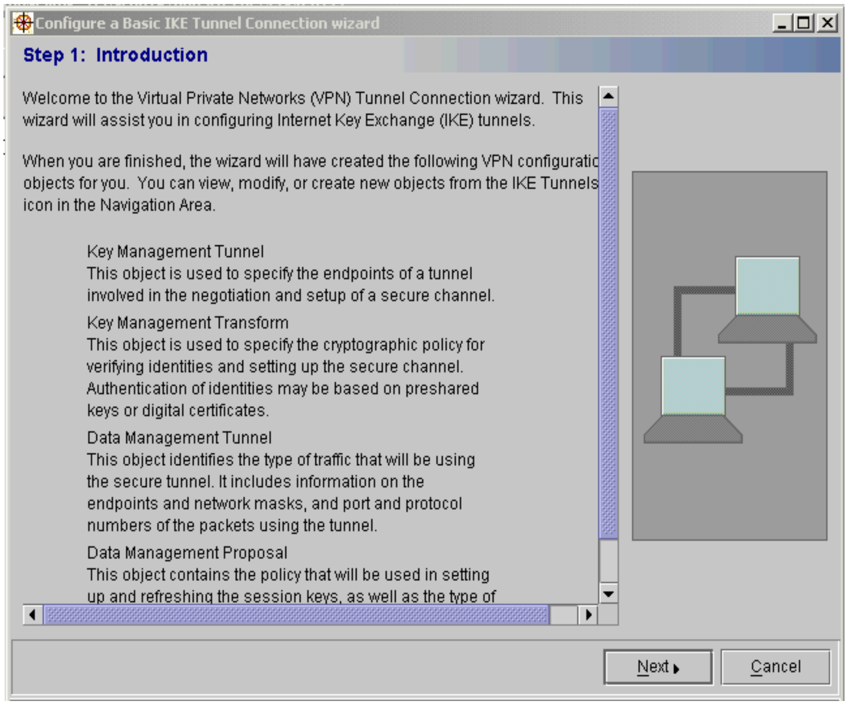
`http://www.rsasecurity.com/rsalabs/pkcs/pkcs-11/`

10.2.27.4 IP Key Encryption Security

The Internet Key Exchange protocol to provide Virtual Private Netoworking (VPN) support in AIX 5L has been enhanced to enable the use of Certificate Revocation Lists (CRL) when authenticating remote users or devices. This is an important improvement in improving scalability of VPNs through the use of Digital Certificates for a large number of users. When CRLs are used, digital certificates provide credentials for authentication, and individual users may be revoked by specifying their certificate number to the CRL. CRLs may be fetched through HTTP or LDAP using socks4 or socks5 protocol.

The user interface on AIX 5L Web-based System Manager for setting up tunnels has been streamlined and simplified. A full-function wizard guides the user through initial IKE tunnel definition as shown in Figure 139 on page 433. Policy information has been reorganized to make IP Security tunnel configuration more intuitive and require fewer steps. To use this function, you have to get the following filesets installed on your system:

- bos.net.ipsec.rte
- bos.net.ipsec.keymgt
- bos.net.ipsec.websm



Step 1: Introduction

Welcome to the Virtual Private Networks (VPN) Tunnel Connection wizard. This wizard will assist you in configuring Internet Key Exchange (IKE) tunnels.

When you are finished, the wizard will have created the following VPN configuratic objects for you. You can view, modify, or create new objects from the IKE Tunnels icon in the Navigation Area.

Key Management Tunnel
This object is used to specify the endpoints of a tunnel involved in the negotiation and setup of a secure channel.

Key Management Transform
This object is used to specify the cryptographic policy for verifying identities and setting up the secure channel. Authentication of identities may be based on preshared keys or digital certificates.

Data Management Tunnel
This object identifies the type of traffic that will be using the secure tunnel. It includes information on the endpoints and network masks, and port and protocol numbers of the packets using the tunnel.

Data Management Proposal
This object contains the policy that will be used in setting up and refreshing the session keys, as well as the type of

Figure 139. Configuration a basic IKE Tunnel Connection

Other IKE enhancements include the use of the commit bit to synchronize the use of Security Associations, the definition of policies to simplify the configuration using dynamic IP addresses, or DHCP. System administrator can define a Virtual Private network by one policy and a list of group members. They can also define default policies to specify the security parameters that are to be used when the addresses are dynamically assigned.

IKE support has also been extended to include IP Version 6 protocols. Thus the IP Security functions for AIX 5L include the definition of static filter for IP Version 4 and 6, manually and dynamically defined privated tunnels using IP Security protocol over IP Version 4 and 6 networks.

10.2.27.5 Directory-based Resolvers

In AIX 5L, the name resolver routines have been enhanced to include resolving hostnames through a LDAP (Lightweight Directory Access Protocol) server. The ordering of name resolution services can be specified in any of the following:

- /etc/netsvc.conf file
- /etc/irs.conf file
- `NSORDER` environment variable. For example: `NSORDER=bind,ldap`

Schema defines the rules for ordering data on a LDAP server. The ibm-HostTable object class, the proposed schema, was accepted by the IBM SecureWay Directory product. A new command, `hosts2ldif`, was created to produce an LDIF (LDAP Data Interchange Format) file from `/etc/hosts`. This LDIF file is used to populate the hosts database on the LDAP server. The LDAP client uses `/etc/resolv.ldap` to access the information from the LDAP server.

10.2.28 AIX Stand-alone LDAP directory product

The AIX Stand-alone Lightweight Directory Access Protocol (LDAP) product provides client access to directory data on a server using standard Internet protocols (LDAP and HTTP). LDAP V3.1 is provided with AIX 4.3.3. Previous releases of AIX come with LDAP V2.1

10.2.28.1 What is LDAP?

Before we discuss LDAP, we should be familiar with directories. A directory is a listing of information about objects arranged in some order. It is suitable for storing such information as lists of books, e-mail addresses of people, and so on. A user or a program can search the directory to locate a book or email-address for a specific person. It is, actually, a database, but the differences between general purpose databases and directories are:

- Directories are read-mostly whereas databases tend to change rapidly. Because of the nature of directories that store such static information as phone numbers, the contents do not change very often. On the other hand, the information stored in databases, such as the order quantity of an item, changes rapidly.
- The data consistency requirement to directories is not strict. Because directories store static information, they might not have transaction support for data integrity. Duplicates and out of date data are acceptable.

LDAP itself is a protocol for a client to communicate with LDAP servers that implement an X.500-like directory. LDAP has its origin in the directory access protocol (DAP) of X.500. Because DAP depends on OSI, it has not been adopted widely by commercial environments. LDAP has been developed as a lightweight alternative to DAP. Compared with DAP, LDAP has the following advantages: It runs over TCP/IP, its function model is simpler, and it uses string representation rather than ASN.1.

The LDAP naming model defines how entries are identified and organized. Although this model is just one aspect of LDAP architecture, understanding this model gives you a conceptual view of LDAP and directory. Entries are organized in a tree-like structure called the directory information tree (DIT).

The following are the basic characteristics of LDAP:

- **Client-server model** - LDAP adopts a client-server model. An LDAP client performs protocol operations, such as query or modify, using LDAP API against an LDAP server, and the server returns a response after completing these operations.
- **Distributed directories** - Although this is not a characteristic of LDAP but of LDAP servers, a directory can be distributed. A distributed directory can be partitioned or replicated. When partitioned, an LDAP client can access information not stored in the local directory through the use of a kind of link stored in the local directory. The link points to the location of the information stored in the remote directory. This link is called a referral. When replicated, a client can access information stored in the nearest directory server. LDAP itself has some mechanisms for accessing distributed directories efficiently. LDAP, as its name implies, is essentially optimized for high speed access to the information and the use of the URL as a resource locator helps a human or a program locate information it needs more easily.
- **Security** - LDAP incorporates a security model in its specification. The model, SASL, will be described later. On the LDAP server's side, some kind of ACL mechanisms are typically implemented, although they are not standardized at the time of this writing; some transport level security mechanisms are also incorporated. Mechanisms, such as SSL and TLS, are used to authenticate the client or the server (or both) and the encrypted data transferred between the client and server.

10.2.28.2 LDAP protocol support

AIX Stand-alone LDAP supports Version 3 of the LDAP protocol. There is, currently, no LDAP Version 3 RFC that is approved. This product has been developed using Internet Draft "Lightweight Directory Access Protocol (v3) draft-ietf-asid-ldapv3-protocol-04.txt", which replaces LDAP (v2) RFC 1777.

Both V2 and V3 LDAP clients and servers are supported. The following combinations of clients and servers have been tested:

- AIX Stand-alone LDAP Version 3 client with Netscape Version 2 server

- A University of Michigan Version 2 client with AIX Stand-alone LDAP server
- Netscape Version 2 client with AIX Stand-alone LDAP server

10.2.28.3 Stand-alone LDAP directory server

The key distinguishing feature between the AIX Stand-alone LDAP server implementation and other LDAP server implementations is the use of DB2 as the back-end data store. See Figure 140 on page 437 for the major components included in the server package.

The features included in the AIX stand-alone LDAP product are:

- DB2 back-end
- ODBC Driver Manager and DB2 driver
- RDB Glue
- SLAPD
- Server replication
- Administration utilities
- Administration GUI
- HTTP gateway

Note

The references to Oracle and Oracle driver in Figure 140 on page 437 are merely an example of future possibilities. DB2 is the only supported database in this release.

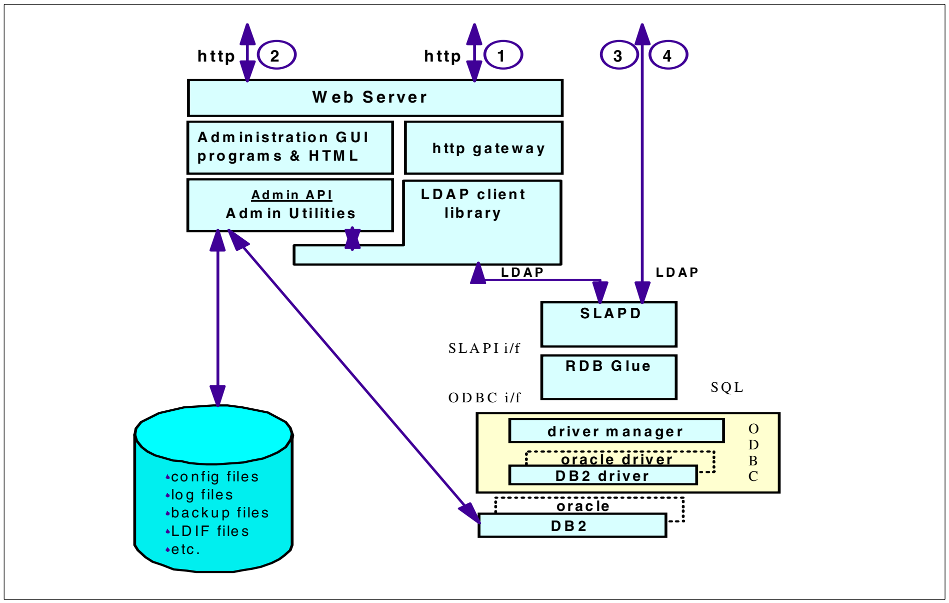


Figure 140. Stand-alone LDAP directory server - Details

DB2 back end

DB2 provides a robust, scalable, industry-tested basis for storage of the directory data. It includes support of Binary Large Objects (BLOBs) that facilitates use of the directory as an efficient object store. The Open DataBase Connectivity (ODBC) interface is used to connect DB2 as the directory back end. Using ODBC as the interface allows for the future inclusion of other relational databases as back-ends.

ODBC

This includes ODBC Driver Manager and the DB2 (ODBC) Driver. The ODBC Driver Manager provides the ODBC API to the LDAP directory server. The DB2 driver plugs into the ODBC framework and connects with the DB2 interfaces. Both of these components of ODBC are provided with the single-user version of DB2 that is shipped with the AIX Stand-alone LDAP Directory product.

RDB Glue

The Relational DataBase (RDB) Glue code ties together two architected interfaces to provide the data store for the directory. The SLAPD component handles incoming LDAP requests and generates calls to a set of APIs defined as the SLAPI interface. RDB Glue provides a matching set of routines that plug into this API, take the previously mentioned API calls, and generate SQL statements in the form required by the ODBC interface to read or write information to DB2.

SLAPD

SLAPD is the portion of the directory server that understands LDAP. It is a multi-threaded daemon that receives client requests, works with the DB2 back-end to process them, and returns the results.

Server replication

SLAPD process threads also monitor the replication log file and pass the corresponding update requests on to the replica server(s).

No shutdown of LDAP server is necessary to copy directory data to initialize a replica server.

HTTP access to directory

An HTTP gateway is provided to allow Web browsers that are not LDAP-enabled to do searches on the directory. The gateway is a cgi-bin program that presents a form to the user, through the browser, to gather the parameters for the search (such as a search base, scope, search filter, and so on). Once the search information has been passed to the gateway program, it acts as an LDAP client generating the requests to do the search and then receiving the results and passing them back to the browser for display to the end-user.

10.2.28.4 Security

The AIX Stand-alone LDAP client and server implementation support SSL (Version 2.0 or higher), an emerging standard for World Wide Web security. SSL provides encryption of data and transport of X.509v3 public-key certificates and revocation lists. The server may be configured to run with or without the SSL support. When the server is configured to support SSL (accepting connections over a secure port - defaults to 636), it still accepts connections from clients that choose not to use SSL (these clients still specify the standard unsecure port - defaults to 389). The LDAP either flows directly over TCP/IP (using the standard sockets interfaces) or over the SSL.

Authentication

The following authentication options are supported:

- No authentication
- Simple authentication (password)
- X.509v3 public-key certificate at the SSL

Note

Kerberos authentication is not supported.

10.2.28.5 LDAP-related RFCs and Internet drafts implemented

The following lists contain Internet drafts and RFCs for LDAP and X.500 implemented in the AIX Stand alone LDAP Directory product.

11.10.1 Internet drafts

Internet drafts may be viewed at the following Web site:

`http://www.internic.net/internet-drafts`

The following are Internet drafts:

- Protocol Definition (March 25, 1997)
- draft-ietf-asid-ldapv3-protocol-04.txt
- Standard and Pilot Attribute Definitions (March 1997)
- draft-ietf-asid-ldapv3-attributes-04.txt
- A String Representation of LDAP Search Filters (March 1997)
- draft-ietf-asid-ldapv3-filter-00.txt (obsolete)
- Replaced by <draft-ietf-asid-ldapv3-filter-02.txt> (May 1997)
- Extensions for Dynamic Directory Services (March 25, 1996)
- draft-ietf-asid-ldapv3ext-03.txt (obsolete)
- Replaced by <draft-ietf-asid-ldapv3ext-04.txt> (May 1997)
- A UTF-8 String Representation of Distinguished Names (March 1997)
- draft-ietf-asid-ldapv3-dn-02.txt (obsolete)
- Replaced by draft-ietf-asid-ldapv3-dn-03.txt (April 1997)
- The LDAP Application Program Interface (October 1996)
- draft-howes-ldap-api-00.txt
- Use of Language Codes in LDAP V3 (March 1997)
- draft-ietf-asid-ldapv3-lang-01.txt
- Definition of an Object Class to Hold LDAP Change (March 25 1997)
- draft-ietf-asid-changelog-00.txt
- LDAP Multi-master Replication Protocol (March 20, 1997)
- draft-ietf-asid-ldap-mult-mast-rep-00.txt

- The LDAP Data Interchange Format (LDIF) (Nov 25, 1996) (March 24 1997)
- draft-ietf-asid-ldif-00.txt

LDAP-Related RFCs

For more information on LDAP and its associated components please refer to the RFCs shown in

Table 23. LDAP-related RFCs

RFC Number	RFC Title
1558	A String Representation of LDAP Search Filters
1738	Uniform Resource Locators
1777 l	Lightweight Directory Access Protocol
1778	The String Representation of Standard Attribute Syntaxes
1779	A String Representation of Distinguished Names
1798	Connectionless LDAP
1823	The LDAP Application Program Interface
1959	An LDAP URL Format
1960	String format of LDAP search filter

10.2.29 Quality of service support

AIX 4.3.3 introduces QoS support. The demand for QoS arises from such applications as digital audio/video applications or real time applications. For more detailed information about QoS, you may refer to section 10.1.1, “Quality of Service (QoS)” on page 361.

10.2.29.1 AIX implementation of QoS

AIX QoS implementation is based on the Internet Engineering Task Force (IETF) standards, Integrated Services (IntServ), and Differentiated Services (DiffServ). IntServ utilizes the Resource ReSerVation Protocol (RSVP) available to applications via the RSVP API (RAPI). DiffServ support includes IP packet marking for IP packets selected via filtering. The AIX QoS also offers bandwidth management functions, such as Traffic Shaping and Policing. The AIX QoS scope covers both QoS and policy-based networking. This enhancement to AIX provides System Administrators with the benefits of both QoS support and policy-based networking in meeting the challenges of QoS offerings across complex networks.

10.2.29.2 New enhancements in AIX 5L

AIX 5L further enhances the QoS implementation to support overlapping policies in the QoS manager. And for the manageability of a QoS configuration, AIX 5L also offers four new commands.

QoS manager overlapping policies

In AIX 5L, the capability to specify priority for a policy is added. This is important when two or more overlapping policies are installed, the policies can be enforced in order of highest policies. The priority for any specific policy can be specified by manually editing the ServicePolicyRules stanzas in the /etc/policyd.conf policy agent configuration file. Alternatively you can use the new command line interface as described in below.

QoS manager command line support

Beginning with AIX 5L four new command line programs will be available to add, modify, delete, or list Quality of Service policies. These AIX commands operate on the /etc/policyd.conf policy agent configuration file. Once you perform one of these commands, the change takes effect immediately and the local configuration file of the policy agent gets updated to permanently keep the change.

Tho QoS command line interface consists of the command provided in the following with their given syntax and usage:

- `qosadd` command adds the specified Service Category or Policy Rule entry in the policyd.conf file and installs the changes in the QoS manager.
- `qosmod` command modifies the specified Service Category or Polity Rule entry in the policyd.conf file and installs the changes in the QoS manager.
- `qoslist` command lists the specified Service Category or Policy Rule.
- `qosremove` command removes the specified Service Category or Policy Rule entry in the policyd.conf file and the associated policy or service in the QoS Manager.

10.2.30 Interface specific network options

AIX implements a set of system-wide network parameters, and some of them can impact the performance of TCP/IP. AIX supports a wide range of physical media on a single system (SLIP, PPP, ISDN, 10 Mbps Ethernet, gigabit ethernet and others) and using the same network parameter for each interface may not be optimal since some of them may have unique characteristics that require special values.

In AIX 4.3.3, the concept of interface-specific network options (ISNO) has been introduced. The user has the ability to activate or deactivate ISNO using the `no` command. A new global option, use_isno, is available, and its default value is 1 (enabled). The intended use of this option is for service-related issues; to allow for a diagnostic tool to eliminate potential tuning errors.

This design allows the definition of adapter-specific defaults that are shipped in the predefined attribute ODM database, PdAt. The global network options are still maintained via the `no` command, but the value of the ISNO takes precedence over the global values, assuming that the TCP/IP traffic flows over an interface that has ISNO set.

10.2.31 Cisco EtherChannel support

Etherchannel is an aggregation technology that allows you to combine multiple Ethernet adapters to form a larger pipe. Etherchannel allows a server-to-switch connection throughput of between 40 and 800 Mbps, depending on the settings for the adapters, and, thus, helps address what may be a throughput bottleneck for you. This throughput would be an aggregate over many connections to different machines.

AIX 4.3.3 introduces the support of Cisco EtherChannel, helping to solve network bandwidth problems in an Ethernet environment.

10.2.31.1 How Etherchannel works

The Etherchannel should look exactly like an Ethernet adapter to the upper layers. Any upper layer (IP, SNA, DLPI, to name a few) that can connect to an Ethernet adapter through network services should work over an Etherchannel without any code changes.

The Etherchannel works by having all of the device drivers that are a part of the channel connected to the same switch, as long as it is a switch that support's Cisco's Etherchannel. Traffic is sent on the network over one of the devices that are part of the channel. The destination of the packet should remian the same regardless of which device is used. The adapter to be used is decided by hashing the lower bits of the destination MAC address of the outgoing packet for non-IP traffic. For IP traffic, the lower bits of the destination IP address are used to decide the adapter. Hashing of the addresses provides for the traffic between a particular source and destination to use the same interface or adapter. This avoids packets reaching the destination out of order; some higher layer protocols have problems handling out of order packets.

Likewise, any packet received by one of the drivers should be sent to the user regardless of which device it was received on. It has one MAC address, and, for IP, it has one IP address.

When an adapter of an Etherchannel is down, traffic is rerouted through one of the other adapters; this is transparent to the upper layers. For the incoming traffic, the packet can be received over any of the adapters. Etherchannel provides a scalable server-to-switch bandwidth without having to move to new technology, such as Gigabit Ethernet. In the future, the Etherchannel technology could possibly be used for multiple Gigabit Ethernets. Etherchannel operates at Layer 2, below the protocol stack. The Etherchannel is implemented as a kernel extension and is a pseudo device that attaches itself to the network services (of CDLI) like the other real Ethernet device drivers.

The existing Ethernet adapters can be used to form a channel without any changes to the existing device drivers. The Etherchannel driver does not maintain any statistics; it uses the statistics maintained by the adapters.

10.2.32 Virtual IP Address Support

In previous AIX releases, an application had to bind to a real network interface in order to get access to a network or network services. If the network became inaccessible or the network interface failed, the application’s TCP/IP session was lost, and consequently, the application was no longer available.

AIX 5L offers support for virtual IP addresses (VIPA) for IPv4 and IPv6. The VIPA related code is part of the bos.net.tcp.client fileset which belongs to the BOS.autoi and MIN_BOS.autoi system bundles and therefore will always be installed on your AIX system. With VIPA, the application is bound to a virtual IP address, not a real network interface that can fail. When a network or network interface failure is detected (using routing protocols or other schemes), a different network interface can be used by modifying the routing table. If the re-routing occurs fast enough, the TCP/IP session will not be lost.

This capability allows the system administrator to define a virtual IP address for a host and, from a TCP connection standpoint, decouple the IP address associated with physical interfaces. As a result, the user connection should not be affected if some interfaces go down.

A traditional IP address is associated with a physical network adapter. VIPA is supported by a network interface that is not associated with any particular network adapter. The operating system will interact with a virtual interface through the interface specific device file which is located in the /dev directory. The device name consists of the two letter abbreviation for the virtual interface, vi, and an appended interface number. The VIPA system management tasks are supported by the interface related, high level operating system commands: `mkdev`, `chdev`, `rmdev`, `lsdev`, `lsattr`, `ifconfig`, and `netstat`. Also, all VIPA management tasks are covered by SMIT and the Web-based System Manager.

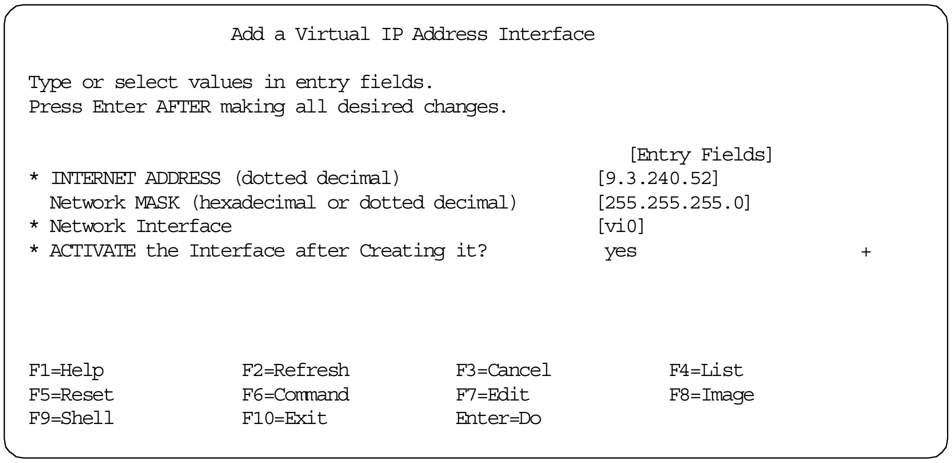
The example below shows how to configure a virtual interface, vi0, for the Internet address 9.3.240.51 with the netmask of 255.255.255.0, using the high level command `mkdev`.The virtual interface belongs to the device class if, the subclass VI, and is of the device type vi.

```
# mkdev -c if -s VI -t vi -a netaddr=’9.3.240.52’ -a etmask=’255.255.255.0’
-w ‘vi0’ -a state=’up’
```

Then by using the `lsdev -Ccif` command, you can identify if the virtual interface is created as following:

```
# lsdev -Ccif
en0  Defined    Standard Ethernet Network Interface
et0  Defined    IEEE 802.3 Ethernet Network Interface
lo0  Available  Loopback Network Interface
tr0  Available  Token Ring Network Interface
cti0 Defined    Configured Tunnel Interface
vi0  Available  Virtual IP Address Network Interface
```

Or you can use SMIT fastpath mkinetvi (smit mkinetvi command) to configure a virtual interface as shown below screen.



```
                  Add a Virtual IP Address Interface

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* INTERNET ADDRESS (dotted decimal)                 [9.3.240.52]
  Network MASK (hexadecimal or dotted decimal)      [255.255.255.0]
* Network Interface                                 [vi0]
* ACTIVATE the Interface after Creating it?          yes                  +




F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Virtual interface vs. real network interface

As you can see in the prior example, virtual network interfaces are similar to traditional network interfaces in most ways. Table 24 shows the similarity and difference between Virtual interface and traditional network interface.

Table 24. Virtual interface vs. traditional network interface

Similarities	Differences
- It is configured in the same way, using `ifconfig`, `mkdev` or system management tools. - Multiple virtual interfaces can be configured and a virtual interface can have aliases. - Applications can bind to a virtual interface just like any other network interface.	- The first virtual interface becomes the preferred source address when an application (except telnet) is locally bound to a wildcard address. - Packets never go in or out of the virtual interface. The packet count for the virtual interface will always be zero. - No route can point to a virtual interface; the virtual interface does not have an interface route. - The virtual interface does not respond to ARP requests.

Data and control flow for network traffic through a virtual interface

When an application is locally bound (IP layer) to a wildcard address connects to a remote host, a VIPA is selected as its source address. The interface the outgoing packet uses is determined by the route table and is based solely on the destination address. The remote host receives the packet and then tried to send a response to the host using the virtual address. The remote host (and all routers along the way) must have a route that will send the packet with the virtual address to one of the network interfaces of the host with a virtual address.

The either gated daemon running on the host with VIPA will send information, which enables the adjacent routers and the remote host to add a host route for the virtual address, or the intermediate routes have to be configured manually along the route.

10.2.33 Network Buffer Cache dynamic data support

The Network Buffer Cache (NBC) was introduced in AIX Version 4.3.2. to improve the performance of network file servers, such as the Web server, FTP server, and SMB server. In AIX Version 4.3.3, the NBC design was improved to allow the use of 256 MB private memory segments for caching additional data.

With the same AIX release, a second key for the cache access mechanism was introduced to support the HTTP GET kernel extension with the Fast Response Cache Architecture (FRCA).

AIX 5L further enhances the Network Buffer Cache kernel extension with following features:

- Dynamic Buffer Cache
- Expiration time per cache object

  - time-to-live specified by creator

  - Stored in units of seconds
- Improves performance of dynamic data
- Effective use of memory buffers allows small (1 to 5%) improvement in capacity

There are two ways for an application to exploit the NBC feature:

- Using the send_file() system call.
- Using the Fast Response Cache Architecture (FRCA) API.

The new AIX 5L NBC enhancements are only accessible for applications through the FRCA API.

10.2.34 HTTP Get kernel extension

Starting with AIX Version 4.3.2, the Fast Response Cache Architecture (FRCA) with the HTTP GET kernel extension was introduced to AIX.

AIX 5L improves the FRCA HTTP GET kernel extension to support HTTP 1.1 persistent connections. Other enhancements to the HTTP GET kernel extension include an external 64-bit ready API and additional support for a new cache type based on memory buffers.

10.2.34.1 HTTP 1.1 persistent connections support

When AIX 4.3.2 was released, HTTP Version 1.0 was the predominant protocol in use with a major part of all requests referring to static content. Since then, a shift toward HTTP Version 1.1 has taken place. One of the major difference between the two versions of HTTP is the new version's well-defined ability to handle multiple requests per connection while the old version almost always closes a connection after a single request. Keeping a connection established for several requests allows the underlying transport layer protocol (TCP) to make better use of the available bandwidth by adapting to it over time.

10.2.34.2 External 64-bit FRCA API

Starting with AIX 5L, an external 64-bit FRCA API is supported to allow more user space applications to exploit the existing function of the HTTP GET kernel extension.

The external API largely follows the structure of the internal API that consists of a set of functions to create and control an FRCA instance and another set of functions that create and fill a cache for a given FRCA interface.

The services compose the external API; it defined in /usr/include/net/frca.h. They are made available to user space applications through the libfrca.a library:

FrcaCtrlCreate Creates an FRCA control instance.

FrcaCtrlDelete Deletes an FRCA control instance.

FrcaCtrlStart Starts the interception of TCP data connections for a previously configured FRCA instance.

FrcaCtrlStop Stops the interception of TCP data connection for a FRCA instance.

FrcaCtrLog Modifies the behavior of the logging subsystem.

FrcaCacheCreate Creates a cache instance within the scope of an FRCA instance.

FrcaCacheDelete Deletes a cache instance within the scope of an FRCA instance.

FrcaCacheLoadFile Loads a file into a cache associated with an FRCA instance.

FrcaCacheUnloadFile Removes a cache entry from a cache that is associated with an FRCA instance.

10.2.35 TCP/IP routing subsystem enhancements

AIX 5L offers multipath routing and Dead Gateway Detection (DGD) as new features of the TCP/IP routing subsystem.

10.2.35.1 Multipath routing

Prior to AIX 5L, a route had to be unique and it was identified by its destination, netmask, and group ID. With the new multipath routing feature in AIX 5L, routes no longer need to have a different destination, netmask, or group ID list. If there are several routes that equally qualify as a route to a destination, AIX will use a cyclic multiplexing mechanism (round-robin) to choose between them. The benefit of this feature is two fold:

- Enablement of load balancing between two or more gateways.
- Feasibility of load balancing between two or more interfaces on the same network can be realized. (The administrator would simply add several routes to the local network, on through each interface.)

In order to implement multipath routing, AIX 5L allows you to define a user-configurable cost attribute for each route and offers the option to associate a particular interface with a given route. These enhancements are configurable by the parameters -hopcount and -if of the `route` command. The syntax and usage for the `route` command is documented in the AIX command reference.

User-configurable cost attribute of routes

The user-configurable cost of a route is specified as a positive integer value for the variable associated with the -hopcount parameter. The integer can be any number between 0 and the maximum possible value of MAX_RT_COST. MAX_RT_COST is defined in the /usr/include/net/route.h header file to be the value of INT_MAX. The value of INT_MAX is defined in /usr/include/sys/limits.h to be 2147483647. The header files will be on your system after you installed bos.adt.include file set.

10.2.35.2 Dead Gateway Detection

AIX 5L implements Dead Gateway Detection (DGD) based on the requirements given in RFC 1122 section 3.3.1.4 and 3.3.1.5, and RFC 816. These RFCs contain a number of suggestions on mechanisms for doing DGD, but no completely satisfactory algorithm has been identified. In particular, the RFCs require that pinging to discover the state of a gateway be avoided (or extremely limited). They recommend that the IP layer receive hints that a gateway is up or down from transport and other layers that may have some knowledge of whether a data transmission succeeded.

There are two possible modes in DGD: active mode and passive mode. However, in active mode, the AIX 5L DGD status information of a gateway is collected with the help of pinging and hence the AIX 5L DGD implementation is not fully compliant with the RFCs mentioned above.

Passive Dead Gateway Detection

Passive mode will work without actively pinging the gateways known to a given system. Passive DGD will use a backup route if a dysfunctional gateway has been detected. The backup route can have a higher current cost than the route associated with the dysfunctional gateway which allows to configure primary (lower cost) gateways and secondary (higher cost) backup gateways. As such DGD expands the TCP inherent failover between alternate equal cost routes as introduced in Multipath routing section.

The passive DGD mechanism depends on protocols which provide information about the state of the relevant gateways. If the protocols in use are unable to give feedback about the state of a gateway, a host will never know that a gateway is down and no action will be taken.

Active Dead Gateway Detection

Passive Dead Gateway Detection has low overhead and is recommended for use on any network that has redundant gateways. However, passive DGD is done on a best-offer basis only. Some protocols, such as UDP, do not provide any feedback to the host if data transmission is failing and, in this case, no action can be taken by passive DGD.

A new network option called dgd_ping_time allow the system administrator to configure the time interval between the periodic ICMP echo request/reply exchanges (ping) in units of seconds. The network option dgd_ping_time can be displayed and changed by the `no -o` command and is set to 5 seconds by default.

The `no` command output shows the value for dgd_ping_time on a system where this specific network option is set to the default of 5:

```
# no -o dgd_ping_time

dgd_ping_time = 5
```

Active Dead Gateway Detection will be off by default and we recommended that it be used only on machines that provide critical services and have high availability requirements. Since active DGD imposes some extra network traffic, network sizing, and performance issues, enabling active DGD should receive careful consideration. This especially applies to environments with a large number of machines connected to a single network.

DGD network options and command changes

Four new network options are defined for Dead Gateway Detection and all of them are runtime attributes that can be changed at any time. Table 25 gives details of the attributes of these options:

Table 25. Network options for Dead Gateway Detection

Network Option	Default	Description
dgd_packets_lost	3	Specifies how many consecutive packets must be lost before Dead Gateway Detection decides that a gateway is down.
dgd_ping_time	5	Specifies how may seconds should pass between pings of a gateway by active Dead Gateway Detection.
dgd_retry_time	5	Specifies how many minutes a route's cost should remain raised once it has been raised by Passive Dead Gateway Detection. After this time passes, the route's cost is restored to its user-configured value.
passive_dgd	0	Specifies whether Passive Dead Gateway Detection is enabled. A value of 0 turns it off, and a value of 1 enables it for all gateways in use.

If the customized DGD network attributes are intended to be permanent, the system administrator must include the appropriate `no` command in /etc/rc.net. Otherwise, the customized network options will be reset to their default during a system boot. For example, if you like to turn on passive DGD permanently, you have to add the following line in /etc/rc.net:

```
# The following no command enables passive Dead Gateway Detection

# after each system boot

if [ -f /usr/sbin/no ] ; then
```

```
/usr/sbin/no -o passive_dgd=1

fi
```

10.2.36 Packet Capture library

As a packet capture system, AIX has offered the Berkeley Packet Filter (BPF). Additionally, AIX 5L introduces a Packet Capture Library (libpcap.a), which provides a high-level user interface to the BPF packet capture facility. The AIX 5L Packet Capture Library is implemented as part of the libpcap library, Version 0.4 from LBNL (Lawrence Berkeley National Laboratory).

The Packet Capture Library user-level subroutines interface with the existing BPF kernel extensions to allow users access for reading unprocessed network traffic. By using the new subroutines of this library, users can write their own network monitoring tool.

For packet capture, follow this procedure:

1. Decide which network device will be the packet capture device. Use the pcap_lookupdev subroutine to do this.
2. Using the pcap_open_live subroutine, you obtain a packet capture descriptor.
3. Choose a packet filter. The filter expression identifies which packets you want to capture.
4. Compile the packet filter into a filter program using the pcap_compile subroutine. The packet filter expression is specified in an ASCII string. Refer to Packet Capture Library Filter Expressions for more information.
5. After a BPF filter program is compiled, notify the packet capture device of the filter using the pcap_setfilter subroutine. If the packet capture data is to be saved to a file for processing later, open the previously saved packet capture data file, known as the savefile, using the pcap_dump_open subroutine.
6. Use the pcap_dispatch or pcap_loop subroutine to read the captured packets and call the subroutine to process them. This processing subroutine can be the pcap_dump subroutine (if the packets are to be written to a savefile) or some other subroutine you provide.
7. Call the pcap_close subroutine to cleanup the open files and deallocate the resources used by the packet capture descriptor.

Note

The current implementation of the libpcap library applies to IP Version 4 and only the reading of packets is supported. Applications using the Packet Capture Library subroutins must be run as root user.

The files generated by libpcap applications can be read by `tcpdump` and vice-versa. However, the `tcpdump` command in AIX 5L does not use the libpcap library.

The Packet Capture Library libpcap.a will be located in the /usr/lib directory after you have optionally installed bos.net.tcp.server fileset. The bos.net.tcp.server fileset also provides BPF kernel extension (/usr/lib/drivers/bpf), which is used by the libpcap subroutines. The library related header file pcap.h can be examined in the /usr/include/ directory if you install the bos.net.tcp.adt file set. The libpcap sample code, which is also part of the bos.net.tcp.adt fileset, can be found in /usr/samples/tcpip/libpcap.

10.2.37 Firewall Hook

The AIX TCP/IP stack provides a way for other kernel extensions to insert themselves into the stack at specific points using hooks.

AIX 5L introduces two new firewall hooks which expand the functional spectrum of the already existing hooks for IP filtering and offers additional potential to improve the performance of firewalls.

The firewall hook routines provide kernel-level hooks for IP packet filtering enabling IP packets to be selectively accepted, rejected, or modified during reception, transmission, and decapsulation. These hooks are initially NULL, but are exported by the netinet extension and will be invoked if assigned non-NULL values.

The following routines are included in AIX 5L as hooks for IP packet filtering:

- ip_fltr_in_hook
- ip_fltr_out_hook
- inbound_fw (new in AIX 5L)
- outbound_fw (new in AIX 5L)

About the syntax of routines, you can refer to chapter 1 of *Kernel and Subsystem Technical Reference, Volume 1.* which can be found in the AIX 5L manuals at the following URL:

http://www.ibm.com/servers/aix/library/index.html

The ip_fltr_in_hook routine is used to filter incoming IP packets, the ip_fltr_out_hook routine filters outgoing IP packets, and the ipsec_decap_hook routine filters incoming encapsulated IP packets.

The new AIX 5L inbound_fw and outbout_fw firewall hooks allow kernel extension to get control of packets at the place where IP receives them. The outbound_fw hook was added exactly at the point where IP is entered when transmitting packets and the inbound_fw hook at the point where IP is called to process receive packets.

10.3 Windows 2000 networking

This section introduces Windows 2000 networking components that make Windows 2000 a highly communicative operating system. An overview of the Windows 2000 network architecture is given in this section as well as details on the implementation of some network components.

10.3.1 Windows 2000 network architecture

Windows 2000 networking components can be organized into three main categories: file system drivers, transport protocols, and network adapter card drivers. These components communicate with each other through programming interfaces called boundary layers. Figure 141 on page 454 is a representation of the Windows 2000 networking model showing the different network layers as well as the boundary layers. The OSI networking model is super imposed in order to show the relationship between the two.

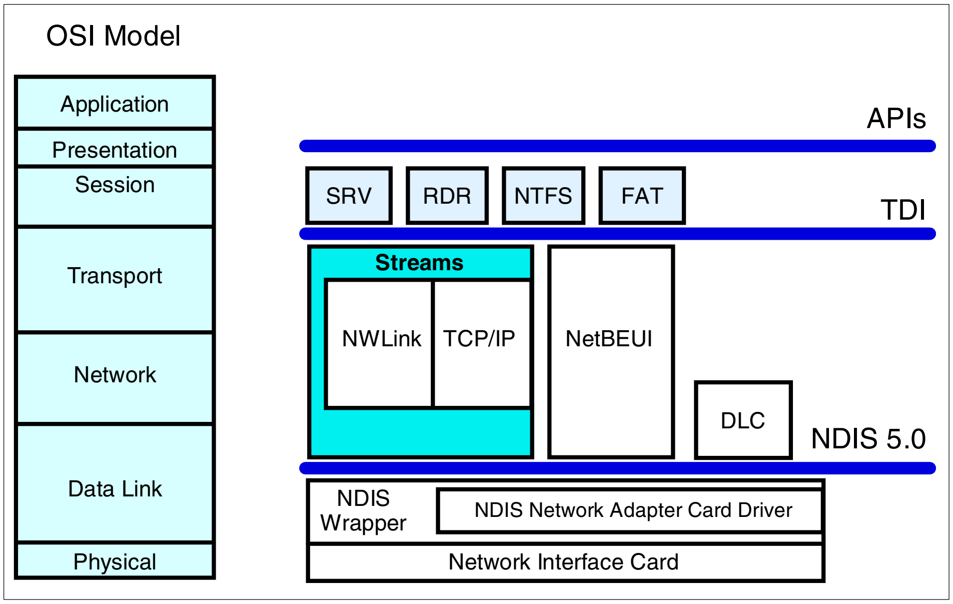


Figure 141. WIndows 2000 Networking Model

In the following sections, you will find descriptions of the different Windows 2000 network layers starting from the lower layers and proceeding to the upper layers.

10.3.1.1 The NDIS interface

Microsoft networking protocols use the Network Device Interface Specification (NDIS) to communicate with network card drivers. Much of the OSI model link layer functionality is implemented in the protocol stack. The current level of NDIS is 5.0.

NDIS can power up or down network adapters when the system requests a power level change. Either the user or the system can initiate this request. For example, the user may want to put the computer in sleep mode, or the system may request a power level change based on keyboard or mouse inactivity. In addition, disconnecting the network cable can initiate a power-down request, provided that the network interface card (NIC) supports this functionality. In this case, the system waits for a configured time period before powering down the NIC because the disconnect could be the result of temporary wiring changes on the network rather than the disconnection of a cable from the network device itself.

NDIS power management policy is not network activity-based. This means that all overlying network components must agree to the request before the NIC can be powered down. If there are any active sessions or open files over the network, the power-down request can be refused by any or all of the components involved.

The computer can also be awakened from a lower power state, based on network events such as a Wake-on-LAN request.

Windows 2000 TCP/IP provides support for:

- Ethernet (and 802.3 SNAP)
- Token Ring (802.5)
- FDDI
- ATM
- ARCNET
- WAN (Such as ISDN, X.25, and dial-up or dedicated asynchronous lines)

In addition, there are some ATM adapters available for Windows that support LAN emulation and appear to the protocol stack as a media type, such as Ethernet.

Media Sense

Media Sense support was added in NDIS 5.0. It provides a mechanism for the Network Interface Card (NIC) to notify the protocol stack of media connect and media disconnect events. Windows 2000 TCP/IP utilizes these notifications to assist in automatic configuration. For instance, in Windows NT 4.0, when a portable computer was located and DHCP-configured on an Ethernet subnet and then moved to another subnet without rebooting, the protocol stack received no indication of the move. This meant that the configuration parameters became stale and not relevant to the new network. Additionally, if the computer was shut off, carried home, and rebooted, the protocol stack was not aware that the NIC was no longer connected to a network, and, again, stale configuration parameters remained. This could be problematic because subnet routes, default gateways, and so on, could conflict with dial-up parameters. Media Sense support allows the protocol stack to react to events and invalidate stale parameters. For instance, if a computer running Windows 2000 is unplugged from the network (assuming the NIC supports Media Sense), after a damping period implemented in the stack (defaults to 20 seconds), TCP/IP will invalidate the parameters associated with the network that has been disconnected. The IP address(es) will no longer allow sends, and any routes associated with the interface are invalidated.

10.3.1.2 Transport protocols

In the Windows NT and 2000 networking models, transport protocols are located above the NDIS wrapper, divided into the following four sets:

- **NBF (NetBEUI Frame)** - A protocol derived from NetBEUI that provides compatibility with existing LAN Manager, LAN Server, and MS-Net installations.
- **TCP/IP** - A widely used, routable protocol for Local Area Networks as well as Wide Area Networks.
- **NWLink** - Microsoft's implementation of Novell's IPX/SPX protocols for communicating across NetWare networks and with NetWare file servers.
- **DLC (Data Link Control)** - DLC provides an interface to access mainframes and printers attached to networks.

In addition, the AppleTalk protocol stack is included to provide support to Macintosh's networks.

NetBEUI and NBF

NetBEUI stands for NetBIOS Extended User Interface. This is the protocol used by LAN Manager and Windows for Workgroups. It is a very efficient protocol for small networks (a small number of nodes on the network) but not very efficient for large networks. There are several reasons for this.

First of all, NetBEUI will not operate properly if two computers have the same name. This is because NetBEUI uses the computer's name as the network address. This can easily happen in a large network.

Second, since NetBEUI does not map the computer name to a network address, it is not routable. This means that network traffic cannot traverse routers and hence limits the physical and practical size of the network. Since NetBEUI by default uses broadcasts to find resources on the network, when the network gets very large, NetBEUI overhead generates a lot of traffic on the network.

> **Note**
>
> A common problem is differentiating between NetBIOS and NetBEUI. NetBEUI is a transport protocol, and NetBIOS is a programming interface.

NetBIOS was developed by IBM and in Windows 2000, NetBIOS is independent of the transport protocol.

The Windows NT and Windows 2000 implementation of NetBIOS over TCP/IP is referred to as NetBT. NetBT uses the following TCP and UDP ports:

- UDP port 137 (name services).
- UDP port 138 (datagram services).
- TCP port 139 (session services).

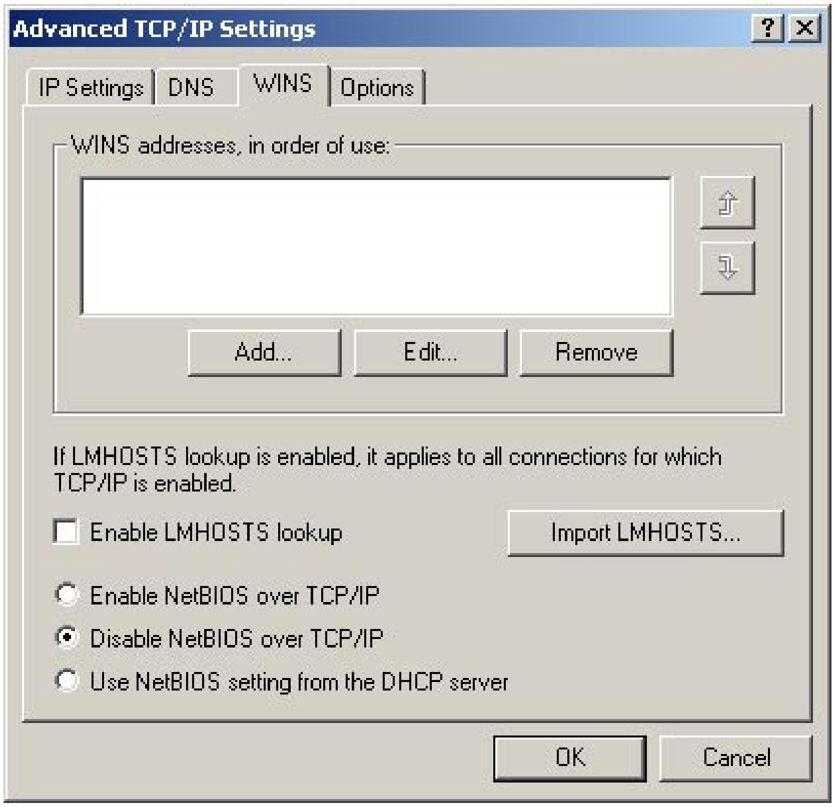
NetBIOS over TCP/IP is specified by RFC 1001 and RFC 1002. The NetBT.sys driver is a kernel-mode component that supports the Transport Driver Interface (TDI) interface. Services, such as the NetLogon, Workstation and Server service, use the TDI interface directly, but traditional NetBIOS applications have their calls mapped to TDI calls by the NetBIOS emulator in the netbios.sys driver. Using TDI to make calls to NetBT is a more difficult programming task, but it provides higher performance and circumvention of some traditional NetBIOS limitations.

Windows 2000 still uses NetBIOS over TCP/IP to communicate with prior versions of Windows NT and other clients, such as Windows 95. However, Windows 2000 is now able to use direct hosting for communication with other Windows 2000 machines.

Direct hosting uses the DNS for name resolution. No NetBIOS name resolution (WINS or broadcast) is used, and the protocol is simpler. Direct Host TCP uses port 445, instead of the NetBIOS TCP port 139.

By default, both NetBIOS and direct hosting are enabled, and both are tried in parallel when a new connection is established. The first to succeed in connecting is used for any attempt. NetBIOS support can be disabled to force all traffic to use direct hosting, shown in Figure 142 on page 458.

Do not change this unless you have a homogenous Windows 2000 environment and do not rely on any legacy NetBIOS applications.



If LMHOSTS lookup is enabled, it applies to all connections for which TCP/IP is enabled.

□ Enable LMHOSTS lookup

○ Enable NetBIOS over TCP/IP

● Disable NetBIOS over TCP/IP

○ Use NetBIOS setting from the DHCP server

Figure 142. Disable NetBIOS over TCP/IP

Netbios names

The NetBIOS name space is flat, meaning that all names within the name space must be unique. NetBIOS names are 16 characters in length. Resources are identified by NetBIOS names, which are registered dynamically when computers boot, when services or applications start, or when users log on. Names can be registered as unique names (one owner) or as group names (multiple owners). A NetBIOS Name Query is used to locate a resource by resolving the name to an IP address.

Microsoft networking components, such as the Workstation and Server services, allow the first 15 characters of a NetBIOS name to be specified by the user or administrator, reserving the 16th character of the NetBIOS name to indicate the resource type (workstation service, server service, master browser, and so on).

NetBIOS name registration and resolution

Windows TCP/IP systems use several methods of locating NetBIOS resources:

- NetBIOS name cache
- NetBIOS name server
- IP subnet broadcasts
- Static Lmhosts file
- Static Hosts file
- DNS servers

NetBIOS name-resolution order depends upon the node type and system configuration. The following node types are supported:

B-node Broadcast node, uses broadcasts for name registration and resolution.

P-node Peer node, uses a NetBIOS Name Server for name registration and resolution.

M-node Mixed node, uses broadcasts for name registration. For name resolution, it tries broadcasts first, but switches to p-node if it receives no answer.

H-node Hybrid node, uses NetBIOS name server for both registration and resolution. However, if no name server can be located, it switches to B-node. It continues to poll for name server and switches back to P-node when one becomes available.

To use a local NetBIOS host file, called Lmhosts, check the appropriate box in the advanced TCP/IP settings as seen in Figure 142 on page 458. This setting uses the local Lmhosts file or WINS proxies plus Windows Sockets gethostbyname calls (using standard DNS and/or local Hosts files) in addition to standard node types.

The only way to change the node type of a client is by using DHCP or to edit the registry of the client

Microsoft ships a NetBIOS name server known as the Windows Internet Name Service (WINS). Most WINS clients are set up as h-nodes; that is, they first attempt to register and resolve names using WINS, and, if that fails, they try local subnet broadcasts. Using a name server to locate resources is generally preferable to broadcasting for two reasons:

- Broadcasts are not usually forwarded by routers

- Broadcasts are received by all computers on a subnet, requiring processing time at each computer

TCP/IP

TCP/IP (Transmission Control Protocol/Internet Protocol) refers to a suite of protocols first developed for the U.S. Department of Defense. It is a widely used, routable transport protocol. It provides connectivity across a wide range of operating systems, such as UNIX systems, and it is a protocol that is very suitable for use in large network LANs or WANs that are linked by routers.

Windows 2000 implements the core of TCP/IP protocols plus a subset of TCP/IP application services. It also provides NetBT (NetBIOS over TCP/IP), which allows use of TCP/IP as a transport layer for NetBIOS and takes advantage of TCP/IP routing capability. NetBIOS over TCP/IP is based on the RFCs 1001/1002.

By default, multi interface Windows 2000 systems do not behave as routers and do not forward IP datagrams between interfaces. However, the Routing and Remote Access Service is included in Windows 2000 Server. It can be enabled and configured to provide full multi protocol routing services.

NWLink

NWLink is Microsoft’s implementation of Novell’s IPX (Internetwork Packet eXchange) network layer and SPX (Sequenced Packet eXchange) transport layer protocol.

NWLink is a transport protocol; it does not allow a Windows 2000 computer to directly access files or printers located on a NetWare server or to act as a file or print server. In order to access resources on a NetWare server, you need to use a redirector, such as the Microsoft Client Service for NetWare or the Novell Client for Windows NT/2000.

Data Link Control (DLC)

DLC is provided for accessing IBM mainframes through 3270 emulation software and for printing to HP printers connected directly to the network. It is not used to communicate between Windows 2000 computers.

The DLC protocol needs to be installed on the computers requiring these types of access routes.

Apple Talk protocol

By using AppleTalk, applications and processes can transfer and exchange data and share resources, such as printers and file servers across a single AppleTalk network or an AppleTalk internet, which is a number of interconnected AppleTalk networks. AppleTalk remote access is supported by the AppleTalk Control Protocol (ATCP).

Transport Driver Interface (TDI)

The TDI is similar in concept to NDIS for the device drivers. It provides a common interface for file systems and I/O manager processes to communicate with the various network transport layers.

The TDI specification describes the set of primitive functions by which transport drivers and TDI clients communicate and the call mechanisms used for accessing them. Currently, TDI is kernel-mode only.

Other protocols

Other transport protocols are provided by third-party vendors, such as DECnet for supporting Digital networks, or the XNS (XEROX Network Systems) transport layer.

10.3.1.3 Session, presentation, and application layers

Above the TDI, you can find the file system drivers, redirectors, and network APIs. Their descriptions follow:

- **File Systems Drivers** - On top of the TDI are the file system drivers. File system drivers allow an application to read and write information to a logical drive. Each type of local file system available in Windows 2000 (such as FAT and NTFS) has its own file system driver to access local partitions. Remote resources, such as network drives, can be accessed in a transparent way by using redirectors.
- **Redirectors** - Windows 2000 provides facilities to share files and printers on the network. These facilities are called the Workstation Service (RDR) and the Server Service (SVR).

  A redirector takes requests from applications, formats the request into a message using an appropriate protocol, such as the Server Message Block (SMB) protocol, and sends it to the server through the network. The server accepts the message, gets the requested data, and sends it back to the requesting computer. Redirectors are implemented as file system drivers and are managed by the I/O Manager. For the I/O Manager, there is no difference between accessing a local file or a remote file.

  In order to access a resource (disk drive, directory, or printer) through the network, you need to specify the name of the resource. There is a Universal Naming Convention (UNC) for resources (servers or sharepoints). UNC names start by following the name of the server and then the name of the shared directory and subdirectories. The syntax of a resource UNC is:

\\computer_name\share_name\subdirectory\filename

- **Network Application Programming Interfaces** - Several APIs are available in Windows 2000 for communicating over the network:
  - NetBIOS
  - Windows Sockets
  - Remote Procedure Calls
  - Network Dynamic Data Exchange (NetDDE)

  Named Pipes and Mailslot are also supported and have been incorporated as an extension into the Windows 2000 operating system.

10.3.2 TCP/IP V4

This section will describe the TCP/IP Version 4 support in Windows 2000.

Figure 143 on page 463 from the Microsoft Web site shows the Windows 2000 TCP/IP network model.

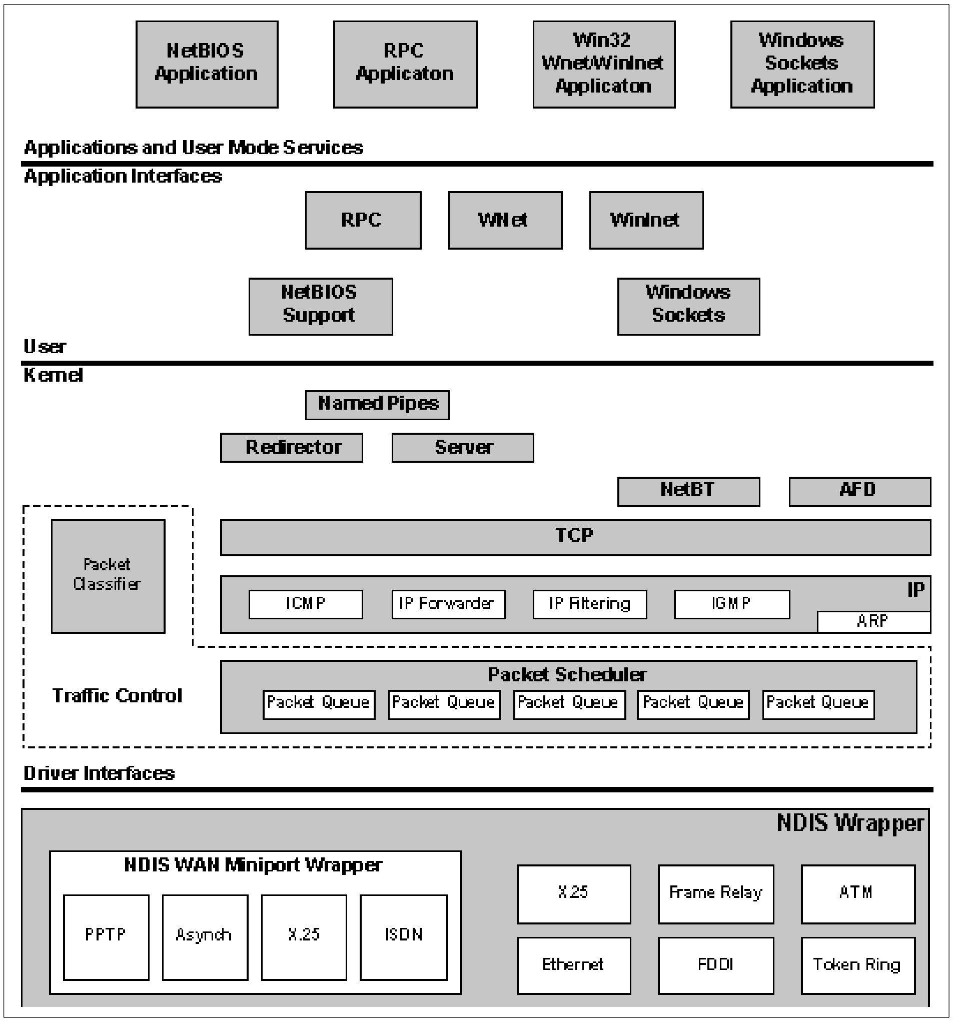


Figure 143. Windows 2000 TCP/IP network model

10.3.2.1 Support for standard features

Windows 2000 supports the following standard features:

- The ability to bind to multiple network adapters with different media types
- Logical and physical multihoming
- Internal IP routing capability
- Internet Group Management Protocol (IGMP) Version 2 (IP Multicasting)
- Duplicate IP address detection
- Multiple default gateways
- Dead gateway detection

- Automatic Path Maximum Transmission Unit (PMTU) discovery
- IP Security (IPSec)
- Quality of Service (QoS)
- ATM Services
- Virtual Private Networks (VPNs)
- Layer 2 Tunneling Protocol (L2TP)
- Performance Enhancements
  - Protocol stack tuning, including increased default window sizes
  - TCP scalable window sizes (RFC 1323 support, off by default)
  - Selective acknowledgments (SACK)
  - TCP Fast Retransmit
  - Round Trip Time (RTT) and Retransmission Timeout (RTO) calculation improvements
  - Improved performance for management of large numbers of connections

Available services

Windows 2000 provides the following services:

- Dynamic Host Configuration Protocol (DHCP) client and service
- Windows Internet Name Service (WINS)
- Dynamic Domain Name Server (DDNS)
- Dial-up (PPP/SLIP) support
- Point-to-Point Tunneling Protocol (PPTP) and Layer 2 Tunneling Protocol (L2TP), used for virtual private remote networks
- TCP/IP network printing (lpr/lpd)
- SNMP agent
- NetBIOS interface
- Windows Sockets Version 2 (Winsock2) interface
- Remote Procedure Call (RPC) support
- Network Dynamic Data Exchange (NetDDE)
- Wide Area Network (WAN) browsing support
- Internet Information Server (http, https, ftp, nntp server)
- Basic TCP/IP connectivity utilities including: finger, FTP, rcp, rexec, rsh, Telnet, and tftp
- Server software for simple network protocols including: Character Generator, Daytime, Discard, Echo, and Quote of the Day
- TCP/IP management and diagnostic tools including: arp, hostname, ipconfig, lpq, nbtstat, netstat, ping, route, nslookup, and tracert

10.3.2.2 RFCs supported by Microsoft Windows 2000 TCP/IP

Requests for Comments (RFCs) are a constantly evolving series of reports, proposals for protocols, and protocol standards used by the Internet community. The following Web site hosts all publicly available RFCs:

`http://www.ietf.org/rfc.html`

RFCs supported by the Windows 2000 implementation of TCP/IP are listed in Table 26.

Table 26. RFCs supported by Windows 2000

RFC	Title
768	User Datagram Protocol (UDP)
783	Trivial File Transfer Protocol (TFTP)
791	Internet Protocol (IP)
792	Internet Control Message Protocol (ICMP)
793	Transmission Control Protocol (TCP)
816	Fault Isolation and Recovery
826	Address Resolution Protocol (ARP)
854	Telnet Protocol (TELNET)
862	Echo Protocol (ECHO)
863	Discard Protocol (DISCARD)
864	Character Generator Protocol (CHARGEN)
865	Quote of the Day Protocol (QUOTE)
867	Daytime Protocol (DAYTIME)
894	IP over Ethernet
919, 922	IP Broadcast Datagrams (broadcasting with subnets)
950	Internet Standard Subnetting Procedure
959	File Transfer Protocol (FTP)
1001, 1002	NetBIOS Service Protocols
1035	Domain Name System (DNS)
1042	A Standard for the Transmission of IP Datagrams over IEEE 802 Networks
1055	Transmission of IP over Serial Lines (IP-SLIP)
1065	Structure and Identification of Management Information for TCP/IP-based internets
1112	Internet Group Management Protocol (IGMP)

RFC	Title
1122, 1123	Host Requirements (communications and applications)
1144	Compressing TCP/IP Headers for Low-Speed Serial Links
1157	Simple Network Management Protocol (SNMP)
1179	Line Printer Daemon Protocol
1188	IP over FDDI
1191	Path MTU Discovery
1256	ICMP Router Discovery Messages
1323	TCP Extensions for High Performance (see the TCP1323opts registry parameter)
1332	PPP Internet Protocol Control Protocol (IPCP)
1518	Architecture for IP Address Allocation with CIDR
1519	Classless Inter-Domain Routing (CIDR): An Address Assignment and Aggregation Strategy
1534	Interoperation Between DHCP and BOOTP
1542	Clarifications and Extensions for the Bootstrap Protocol
1552	PPP Internetwork Packet Exchange Control Protocol (IPXCP)
1661	The Point-to-Point Protocol (PPP)
1662	PPP in HDLC-like Framing
1748	IEEE 802.5 MIB using SMIv2
1749	IEEE 802.5 Station Source Routing MIB using SMIv2
1812	Requirements for IP Version 4 Routers
1828	IP Authentication using Keyed MD5
1829	ESP DES-CBC Transform
1851	ESP Triple DES-CBC Transform
1852	IP Authentication using Keyed SHA
1886	DNS Extensions to support IP version 6
1994	PPP Challenge Handshake Authentication Protocol (CHAP)
1995	Incremental Zone Transfer in DNS

RFC	Title
1996	A Mechanism for Prompt Notification of Zone Changes (DNS NOTIFY)
2018	TCP Selective Acknowledgment Options
2085	HMAC-MD5 IP Authentication with Replay Prevention
2104	HMAC: Keyed Hashing for Message Authentication
2131	Dynamic Host Configuration Protocol
2132	DHCP Options and BOOTP Vendor Extensions
2136	Dynamic Updates in the Domain Name System (DNS UPDATE)
2181	Clarifications to the DNS Specification
2205	Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification
2236	Internet Group Management Protocol, Version 2
2308	Negative Caching of DNS Queries (DNS NCACHE)
2401	Security Architecture for the Internet Protocol
2402	IP Authentication Header
2406	IP Encapsulating Security Payload (ESP)
2581	TCP Congestion Control
2782	A DNS RR for specifying the location of services (DNS SRV)
Draft RFCs	PPP over ISDN; PPP over X.25; Compression Control Protocol

10.3.2.3 PMTU discovery

Windows 2000 introduces the support of PMTU for the Windows operating system. For a more detailed discussion about the PMTU topic, you may refer to section 10.1.2, “Path Maximum Transmission Unit (PMTU)” on page 364.

10.3.3 TCP/IP V6

Support for version 6 of the TCP/IP protocol is still under development by Microsoft but there is an unsupported publicly available version with limited functionality that you can download for evaluation. It is available from Microsoft at:

```
http://msdn.microsoft.com/downloads/sdks/platform/tpipv6.asp
```

Currently, Version 1.4 of the IPv6 stack supports scoped address support in the API and the stack, Plug'n'Play and Power Management, and automated 6to4 configuration.

The scoped address support allows link-local and site-local addresses to be used unambiguously for local communication. Both site-local addressing with site prefixes and multi-sited nodes are supported.

USB and PCMCIA network interfaces can be added to (or removed from) the system on the fly and the stack will reconfigure itself accordingly. Similarly, one can disconnect and reconnect network links or hibernate and resume a system and the IPv6 stack will do the right thing. It is possible to dynamically unload and reload the stack without rebooting.

While still a work-in-progress, support for all of the main features of the IPv6 protocol are in place, in particular the following:

- Basic IPv6 header processing
- Hop-By-Hop and Destination Options headers
- Fragmentation header
- Routing header
- Neighbor Discovery
- Stateless address autoconfiguration
- ICMPv6
- Multicast Listener Discovery (IGMPv6)
- Ethernet and FDDI media
- Automatic and configured tunnels
- IPv6 over IPv4 (Carpenter/Jung draft)
- 6to4 (Carpenter/Moore draft)
- Site-Prefixes (Nordmark draft)
- UDP and TCP over IPv6
- UDP Lite (Larzon draft)
- Raw packet transmission
- Correspondent node mobility
- Router functionality (static routing tables)
- IPSec authentication (AH and ESP, tunnel and transport mode)

There is still not full support for mobility or encryption.

10.3.4 DHCP

DHCP is open and standards-based as defined by IETF Requests for Comments (RFCs) 2131 and 2132. DHCP can automatically configure a host while it is booting on a TCP/IP network as well as change settings while the host is attached. This allows all available IP addresses to be stored in a central database along with associated configuration information, such as the subnet mask, gateways, and address of DNS servers.

10.3.4.1 DHCP enhancements for Windows 2000

For Windows 2000 Server, the Microsoft DHCP server has been enhanced with powerful new features, including:

- Integration of DHCP with DNS
- Enhanced monitoring and statistical reporting for DHCP servers
- New vendor-specific and class ID option support
- Multicast address allocation
- Rogue DHCP server detection
- Windows Clustering for high availability (after IETF release of the server-to-server communications protocol)
- Automatic client configuration

Integration of DHCP with DNS (DDNS)

Domain name system (DNS) servers provide name resolution for network resources and are closely related to DHCP services. For Windows 2000, DHCP servers and DHCP clients may register with DNS. This way, DNS servers are dynamically updated. Of course, this makes life easier for system administrators, who no longer have to manually maintain DNS server configuration files.

Enhanced monitoring and statistical reporting for DHCP servers

Enhanced monitoring and statistical reporting have been added to the DHCP server for Windows 2000. This new feature provides notification when IP addresses are running below a user-defined threshold. For example, an alert could be triggered when 90 percent of IP addresses assigned for a particular scope have been assigned. A second alert can be triggered when the pool of IP addresses is exhausted. To alert network managers, the DHCP scope icon is changed to yellow when the remaining addresses are falling below the defined level and changed to red if the addresses are completely depleted.

The DHCP manager, which supports the Simple Network Management Protocol (SNMP) and Management Information Bases (MIBs), provides a graphical display of statistical data. This helps administrators monitor system status factors, such as the number of available versus depleted addresses or the number of leases being processed per second. Additional statistical information includes the number of messages and offers processed, as well as the number of requests, acknowledgements, declines, negative status acknowledgment messages (Nacks), and releases received.

The total number of scopes and addresses on the server, the number used, and the number available are also viewable. These statistics can be provided for a particular scope or at the server level, which shows the aggregate of all scopes managed by that server.

New vendor-specific and class ID option support

DHCP server for Windows 2000 provides the powerful functionality of allowing vendor-specific options to be defined. These vendor classes are defined by specific vendors and are triggered by data bits that determine whether a given option class is standard or vendor-specific. Once identified as vendor-specific, DHCP looks up the configuration as specified for the specific vendor. This feature enables compelling custom applications for enterprise networks to be introduced quickly. The vendor class and vendor options are described in RFC 2132.

Traditionally, all DHCP clients are treated equally, and the server is unaware of the specific type of clients. This means that the configuration issued by the server must be one that can be common to any DHCP client. User classes allow DHCP clients to differentiate themselves by specifying what type of client they are, such as a desktop or laptop, for example. An administrator can then configure the DHCP server to assign different options, depending on the type of client receiving them.

Multicast address allocation

The Microsoft DHCP server has been extended to allow the assignment of multicast addresses, in addition to unicast addresses. A proposed IETF standard defines multicast address allocation. The proposed standard benefits network administrators by allowing multicast addresses to be assigned in the same fashion as unicast addresses, thus, allowing complete utilization of the existing network infrastructure.

Typical applications for multicast are conferencing or audio, which usually require users to specially configure multicast addresses. Unlike IP broadcasts, which must be readable by all computers on the network, a multicast address is a group of computers using the concept of a group membership to identify those to whom the message is to be sent.

Unauthorized DHCP Server Detection

The Microsoft DHCP server for Windows 2000 is designed to prevent unauthorized DHCP servers from creating address assignment conflicts. This solves problems that might otherwise occur if inexperienced users create unauthorized DHCP servers, assigning improper or unintended IP addresses to clients elsewhere on the network.

The way in which this prevention does work is by the integration of a DHCP server definition with Active Directory. In fact, Active Directory is now used to store records of authorized DHCP servers. When a DHCP server comes up, the Active Directory is used to verify the status of that server. If that server is unauthorized, no response is returned to DHCP requests.

Of course, this does not stop users from installing DHCP server on other operating systems or even stand alone Windows 2000 servers.

Windows Clustering for high availability

Windows Clustering allows two to four servers to be managed as a single system and can be used for servers requiring high availability. By automatically detecting the failure of an application or server and restart it on a surviving server; users would only experience a momentary pause in service.

Automatic client configuration

A change of behavior in the Windows 2000 DHCP client is the ability to automatically configure an IP address and subnet mask when the client is started on a small private network where no DHCP server is available to assign addresses.

If a Microsoft TCP/IP client is installed and set to dynamically obtain TCP/IP protocol configuration information from a DHCP server, rather than being manually configured with an IP address and other parameters, the DHCP client service is engaged each time the computer is restarted.

The DHCP client service now uses a two-step process to configure the client with an IP address and other configuration information:

1. When the client is installed, it attempts to locate a DHCP server and obtain a configuration from it. Many TCP/IP networks use DHCP servers that are administratively configured to hand out information to clients on the network.
2. If this attempt to locate a DHCP server fails, the Windows 2000 DHCP client auto configures its stack with a selected IP address from the IANA-reserved class B network 169.254.0.0. with the subnet mask 255.255.0.0. The DHCP client tests (using an ARP) to make sure that the IP address that it has chosen is not already in use. If it is in use, it selects another IP address (it does this for up to 10 addresses). Once the DHCP client has selected an address that is not in use, it configures the interface with this address. It continues to check for a DHCP server in the background every five minutes. If a DHCP server is found, the

autoconfiguration information is abandoned, and the configuration offered by the DHCP server is used instead.

If the DHCP client has previously obtained a lease from a DHCP server, the following modified sequence of events occurs:

1. If the client's lease is still valid (not expired) at boot time, the client tries to renew its lease with the DHCP server. If the client fails to locate a DHCP server during the renewal attempt, it tries to ping the default gateway that is listed in the lease. If pinging the default gateway succeeds, the DHCP client assumes that it is still located on the same network where it obtained its current lease and continues to use the lease. By default, the client attempts to renew its lease in the background when half of its assigned lease time has expired.
2. If the attempt to ping the default gateway fails, the client assumes that it has been moved to a network that has no DHCP services currently available (such as a home network) and auto configures itself as described in the section above. Once auto configured, it continues to try (in the background) to locate a DHCP server every five minutes.

10.3.5 Quality of service (QoS)

Microsoft introduces support for QoS with Windows 2000. For more detailed information about QoS, you may refer to section 10.1.1, “Quality of Service (QoS)” on page 361. The following QoS components are currently included with the Windows 2000 operating system:

- The Generic Quality of Service (GQoS) API, which is a subset of the Winsock 2 API, allows applications to invoke QoS services from the operating system without the need to understand the underlying mechanisms.
- The QoS service provider (SP) responds to requests from the GQoS API. It provides RSVP signaling and QoS policy support.
- The ACS service and SBM protocol for admission control service on shared media with Kerberos also invoke the traffic control mechanisms.

  ACS is a policy service that runs on top of the Windows 2000 Server. It works in conjunction with the Subnet Bandwidth Manager (SBM). Because LANs are layer 2 technologies, they cannot directly participate in RSVP signaling, which is a layer 3 technology. The SBM solves this problem by enabling intelligent devices that reside on the shared network to volunteer their services as a *broker* for the shared network's resources.

  The ACS combines the resource-based admission control functionality of an SBM with policy-based admission control using Active Directory. The

ACS leverages the fact that the SBM (by advertisement of its presence on a shared subnet) is able to insert itself into the RSVP reservation path and can, therefore, affect admission control.

- A traffic control infrastructure includes a packet scheduler and marker for providing traffic control over drivers and network cards that have no packet scheduling features of their own. It also marks packets for diffserv and 802.1p. Windows 2000 traffic control also includes additional mechanisms, such as ISSLOW and ATM.

10.3.6 Network security

Windows 2000 supports the following security protocols:

- **IPSec** - IPSec provides integrity protection, authentication, and (optional) privacy and replay protection services for IP traffic. IPSec can be used in two modes; transport mode which secures an existing IP packet from source to destination, and tunnel mode which puts an existing IP packet inside a new IP packet that is sent to a tunnel end point in the IPSec format. Both transport and tunnel mode can be encapsulated in ESP or AH headers.
  - IP protocol 50 called the Encapsulating Security Payload (ESP) format, which provides privacy, authenticity, and integrity.
  - IP protocol 51 called the Authentication Header (AH) format, which only provides integrity and authenticity for packets, but not privacy
- **PPP** - PPP encapsulates IP, IPX, and NetBEUI packets within PPP frames and then transmits the PPP-encapsulated packets across a point-to-point link. Typically, most implementations of PPP provide limited authentication methods, such as Password Authentication Protocol (PAP), Challenge Handshake Authentication Protocol (CHAP), and Microsoft Challenge Handshake Authentication Protocol (MSCHAP).
- **PPTP** - PPTP provides authenticated and encrypted communications between a client and a gateway or between two gateways (without requiring a public key infrastructure) by using a user ID and password.
- **L2TP/IPsec** - This protocol places L2TP as payload within an IPSec packet. L2TP encapsulates Point-to-Point Protocol (PPP) frames to be sent over IP, X.25, frame relay, or asynchronous transfer mode (ATM) networks. When configured to use IP as its transport, L2TP can be used as a VPN tunneling protocol over the Internet. L2TP over IP uses UDP port 1701 for tunneling communication.

For a more detailed description of these protocols and security over the network, you may refer to section 10.1.5, “Security over the network” on page 367.

10.3.7 Name resolution

In addition to the RFCs 1001/1002, other mechanisms can be used in Windows 2000 to resolve host names to IP addresses.

10.3.7.1 TCP/IP Hosts File

TCP/IP uses a flat file called HOSTS to map TCP/IP host names to IP addresses and vice versa. This file is located in the %SystemRoot%\system32\drivers\etc directory.

Its AIX equivalent is the /etc/hosts file.

Although this works well on small networks, it is not suitable in large networks where using a name server would facilitate IP address administration.

10.3.7.2 LMHOSTS

This file is similar in concept to the HOSTS file. However, its purpose is to explicitly map NetBIOS names with IP addresses. Usually, this file contains a list of computers that cannot directly be reached via NetBIOS (usually remote computers located on a different network).

This file is %SystemRoot%\system32\drivers\etc\lmhosts.

The format of the LMHOSTS file is very similar to the format of the HOSTS file. This file is strictly reserved for mapping NetBIOS names to IP addresses. It cannot be used instead of the HOSTS file, which is a TCP/IP host database.

As with the HOSTS file, each computer must maintain its own LMHOSTS file. In a large network, administrating LMHOSTS files can be a tedious work. In order to limit the number of files maintained over the network, it is possible to share LMOSTS files at, for example, a department level.

A better solution to the LMHOSTS files administration issue is using a name service called Windows Internet Name Service (WINS), which is provided by Windows 2000 Server.

10.3.7.3 Windows Internet Name Service (WINS)

WINS provides a distributed database for registering and querying dynamic NetBIOS name-to-IP address mapping in a routed network environment. This support for dynamic registering of NetBIOS computer names means that WINS can be used with Dynamic Host Configuration Protocol (DHCP) services to provide easy configuration and administration of Windows-based TCP/IP networks. The WINS server solves the problems inherent in resolving NetBIOS names through IP broadcasts and frees network administrators from the demands of updating static mapping files, such as LMHOST files.

Widows 2000 server is shipped with an enhanced version of WINS. The new implementation of WINS provides a number of features including:

- **Persistent connections** - This configurable feature allows each WINS server to maintain a persistent connection with one or more replication partners to eliminate the overhead of opening and terminating connections and to increase the speed of replication.
- **Manual tombstoning** - Use of the Manual tombstoning feature marks a record for deletion so that the tombstone state for the record is replicated across all WINS servers, thus, preventing an undeleted copy of the record on a different server database from being repropagated. In earlier versions of WINS, the removal of unwanted records could be difficult. Records were marked for removal on only one server, and that information is replicated to other WINS replication partners. Depending upon replication configuration, record removal might not occur correctly
- **Improved management tools** - The WINS Manager is fully integrated with the Microsoft Management Console (MMC) providing a more user-friendly and powerful environment for viewing and managing WINS information.
- **Enhanced filtering and record searching** - These functions help locate records of interest by showing only those that fit a specific criteria. This is particularly useful for analyzing very large WINS databases.
- **Dynamic record deletion and multi-select** - Managing the WINS database is made easier with dynamic record deletion and multi-select. Dynamic and static records can be deleted, and the point-and-click interface makes it possible to delete files with non-alphanumeric characters that could not be handled from the command line.
- **Record verification and version number validation** - Two tools for quickly checking the consistency between various WINS servers are available. The tests are done by comparing the IP addresses of a NetBIOS name query returned from different WINS servers or by examining owner address to version-number mapping tables.
- **Export function** - The Export function can be used to place WINS data into a comma-delimited text file that can be imported into Microsoft Excel,

reporting tools, scripting applications, and so on for analysis and reporting.

- **Increased fault tolerance** - Windows 2000 and Windows 98 allow a client to specify more than two WINS servers (up to a maximum of 12 addresses) per interface. The extra WINS server addresses are used only if the primary and secondary WINS servers fail to respond.
- **Dynamic re-registration** - WINS clients can now reregister their NetBIOS name-to-IP address mapping without rebooting the server.

Note

Windows 2000 clients (and Windows clients with the Active Directory upgrade installed) no longer need to use the NETBIOS name space. WINS is still required for downlevel clients to find servers and vice versa. When there are no more downlevel clients in the enterprise, the WINS servers can be turned off.

10.3.7.4 Domain Name System (DNS)

Name resolution via Domain Name System (DNS) is a service provided by Windows 2000. DNS servers provide the ability to centrally manage the host-name resolution to an ip-address. This is very useful for system administrators who, in this way, do not have to maintain configuration host files on each client workstation manually.

10.3.7.5 Dynamic Domain Name System (DDNS)

Windows 2000 introduces support for DDNS, allowing DNS servers to be automatically updated by DHCP servers when assigning IP-addresses to clients. This feature, along with DHCP support, makes IP network administration very easy.

10.3.8 Mail

This section provides an overview of mail support in Windows 2000.

10.3.8.1 Microsoft Windows Messaging

Windows 2000 includes the Microsoft Windows Messaging System, which enables you to send and receive e-mail through Microsoft’s e-mail client, which is also included with Windows 2000. Messaging supports various service providers that enable it to communicate with a variety of message servers. Included with Windows 2000 are service providers for Internet Mail and Microsoft Mail.

Microsoft Mail has been replaced by a product called Microsoft Exchange. Microsoft Exchange is the product that competes with Lotus Notes/Domino. The Microsoft Messaging Client is a light-weight version of the Microsoft Exchange client.

10.3.8.2 Microsoft Exchange Client

Microsoft Exchange Client can use existing network transport protocols to communicate with Microsoft Exchange Server. Designed to integrate fully and transparently with existing networks, Microsoft Exchange Server requires no changes to the client software, routers, or other applications. It fully supports IPX/SPX, NetBIOS, TCP/IP, or AppleTalk for Apple Macintosh networks.

10.3.8.3 Microsoft Exchange Server

Microsoft Exchange Server features client/server architecture with scalable, intelligent server processes running on one or more Windows 2000 Server-based computers. It provides a powerful communications infrastructure and a set of tightly-integrated components that enable users to easily locate, share, and communicate information within and between teams and even between organizations.

10.3.8.4 Lotus Notes/Domino on Windows 2000

An alternate solution for Mail and Messaging on Windows 2000 is Lotus Notes/Domino. Notes is a client/server environment that allows users (or clients) to communicate securely over a Local Area Network or telecommunications link. Notes combines an application development environment, a document database, and a sophisticated messaging system, giving you the power to create custom applications for improving the quality of everyday business processes in areas, such as product development, customer service, sales, and account management.

At time of writing, Lotus Notes 5.0.6a is the latest available version for the Windows platform.

10.3.9 Network Time Protocol (NTP)

Windows 2000 has the ability to synchronize the system clock using the Network Time Protocol (NTP) as defined in RFC 1305. In a Windows 2000 domain environment, it is highly recommended that this feature is used. More information about the NTP protocol and a list of publicly available NTP servers in the Internet can be found on:

`http://www.ntp.org/`

10.3.10 License management

Windows 2000 offers a License Wizard. This tool is useful for administrators to keep track of the software licenses they use for servers and clients.

10.3.11 Internet/intranet software

This section provides an overview of Internet/intranet software available in Windows 2000.

10.3.11.1 Microsoft Internet Explorer

Microsoft Internet Explorer (IE) is the Microsoft preferred Web browser. Version 5 is a part of Windows 2000 and can be upgraded for free over the Internet. At time of writing, the latest version of IE is 5.5 SP1.

IE version 5 provides the following features:

- **ActiveX Controls** - ActiveX Controls is made of software components that run in Web pages and provide interactive and user-controllable functions. This enables users to view and interact with animation, audio, and video without opening separate programs. Also, ActiveX Controls can be used in applications and written in many popular programming languages including Java, Visual Basic, and Visual C++. Anyway, keep in mind that you must take care when enabling ActiveX on your browser because it might have some Internet security weakness.

  ActiveX Control Pad is a utility that makes it easier to create Web pages that incorporate ActiveX Controls and ActiveX Scripting.
- **ActiveX Scripting** - ActiveX Scripting supports any popular scripting language including VBScript and JScript. Scripts can be used to integrate the behavior of several ActiveX Controls and/or Java applets from the browser or server, thus, extending their functionality.
- **ActiveX Documents** - ActiveX Documents enable you to open a program, complete with its own toolbars and menus, in Internet Explorer. This means you can open non-HTML documents, such as Microsoft Excel or Microsoft Word files, through a Web browser.
- **Dynamic Data Exchange (DDE) Support**
- **HTML and Stylesheet Support**
- **Internet Mail and News** - Internet Mail and News lets you send and receive mail quickly on the Internet and subscribe to your favorite newsgroups with its newsreader. Its complete integration with Internet Explorer lets you check your favorite newsgroups or send mail while you are running Internet Explorer. It provides full support for Internet standards

including SMTP/POP3 and MIME Spell-checking of mail, and it provides full support for Internet NNTP standards.

- **Java Support** - Java Virtual Machine enables any ActiveX-supported browser, such as Internet Explorer, to run Java applets and integrate Java applets with ActiveX Controls.
- **Multimedia** - Internet Explorer comes fully loaded with support for all standard video and audio formats including Audio-Video Interleaved (AVI), QuickTime, MPEG video, WAV, AU, AIFF, and MPEG audio using the Microsoft ActiveMovie technology. You no longer need to download other helper applications to enjoy multimedia on the Internet.

  ActiveMovie provides a universal playback mechanism for video or audio streams through an architecture that exposes all elements of the media stream.
- **NetMeeting** - NetMeeting adheres to major international standards from both the International Telecommunications Union (ITU) and the Internet Engineering Task Force to guarantee broad interoperability between applications and across platforms. For example, you can share a whiteboard between different applications on different platforms thanks to support for standard protocols, such as the ITU's T.120.

  Additional standards supported by Internet Explorer include G.723, H.323, POP3, HTML, and MIME; so, unlike proprietary competitive solutions, NetMeeting can interoperate with solutions from Intel, Ptel, Vtel, Confertech, MCI, ATT, Apple, and other vendor solutions.

  NetMeeting also supports *multipoint* connections, which allow more than two people to join in conversations. Voice conferencing is currently limited to connections between two points, but data conferencing can occur between up to five points, and even more simultaneous users can conference through network-based conferencing services. It includes a tool for connecting to and scheduling the use of network-based conferencing services.
- **Microsoft Chat** - Microsoft Chat is one of the most popular IRC tools available today. With Microsoft Chat, you may see all the available rooms, choose to join even more than one at the same time, and easily chat with the whole room or talk privately with only one partner.
- **Netscape Plug-in Support**
- **Personalization**
- **Security** - Internet Explorer supports Secure Sockets Layer 2.0/3.0 (SSL) and Private Communication Technology 1.0 (PCT). Support for SSL 2.0/3.0 and PCT 1.0 Client authentication lets you present your personal

certificates to Web servers that request it. Personal certificates also make it easier to connect to Web services. Server authentication, “Cookie” Privacy, SOCKS Firewall Support and CryptoAPI, the foundation of the Microsoft Internet Security Framework, provide the underlying security services for secure channels and code signing.

10.3.11.2 Microsoft FrontPage

Microsoft FrontPage is an integrated visual authoring environment for WorldWide Web sites on the Internet or for intranets within organizations. Using FrontPage, any experienced user of typical Windows applications can create, deploy, maintain, and administer a Web site. Microsoft FrontPage makes creating professional-quality Web sites effortless with powerful new functionality, support for the latest Web technologies, and seamless integration with Microsoft Office.

10.3.11.3 Microsoft Internet Information Server (IIS)

Microsoft Windows 2000 Server includes Internet Information Services (IIS) version 5. IIS runs as a service within Windows 2000 and uses other services provided by Windows 2000, such as security and the Active Directory directory service.

10.3.12 Telephony API (TAPI)

TAPI 3.0 is an evolutionary API providing convergence of both traditional PSTN telephony and IP telephony. IP telephony is an emerging set of technologies that enable voice, data, and video collaboration over existing LANs and WANs. TAPI 3.0 enables IP telephony on Microsoft Windows operating systems by providing simple and generic methods for making connections between two or more computers and accessing any media streams involved in the connection.

TAPI 3.0 supports standards-based H.323 conferencing and IP multicast conferencing. It uses the Windows 2000 operating system's Active Directory service to simplify deployment within an organization and includes quality-of-service (QoS) support to improve conference quality and network manageability.

In the past, organizations have deployed separate networks to handle traditional voice, data, and video traffic. Each had different transport requirements, and these networks were expensive to install, maintain, and reconfigure. Furthermore, since these networks were physically distinct, integration was difficult, if not impossible, thus, limiting their potential usefulness.

IP telephony blends voice, video, and data by specifying a common transport, IP, for each, effectively collapsing three networks into one. The result is increased manageability, lower support costs, a new breed of collaboration tools, and increased productivity.

Possible applications for IP telephony include telecommuting, real-time document collaboration, distance learning, employee training, video conferencing, video mail, and video on demand.

TAPI requires some QoS mechanisms in order to guarantee appropriate end-to-end communication in real time. These mechanisms include:

- The Resource Reservation Protocol (RSVP)
- Local Traffic Control:
- Packet Scheduling
- 802.1p
- Appropriate Layer 2 signaling mechanisms
- IP Type of Service and DTR header settings

10.3.13 Network API

Windows 2000 provides a set of services and APIs that allow one to use or develop client/server applications. In Windows 2000, these utilities are referred to as Interprocess Communication (IPC) mechanisms. These mechanisms act at the presentation or application level in the layered network model. As shown in Figure 144 on page 482, the main IPCs provided by Windows 2000 are:

- NetBIOS interface
- Windows sockets
- Named Pipes (NPFS)
- Mailslots (MSFS)
- Remote Procedure Calls (RPCs)
- Network Dynamic Data Exchange (NetDDE)

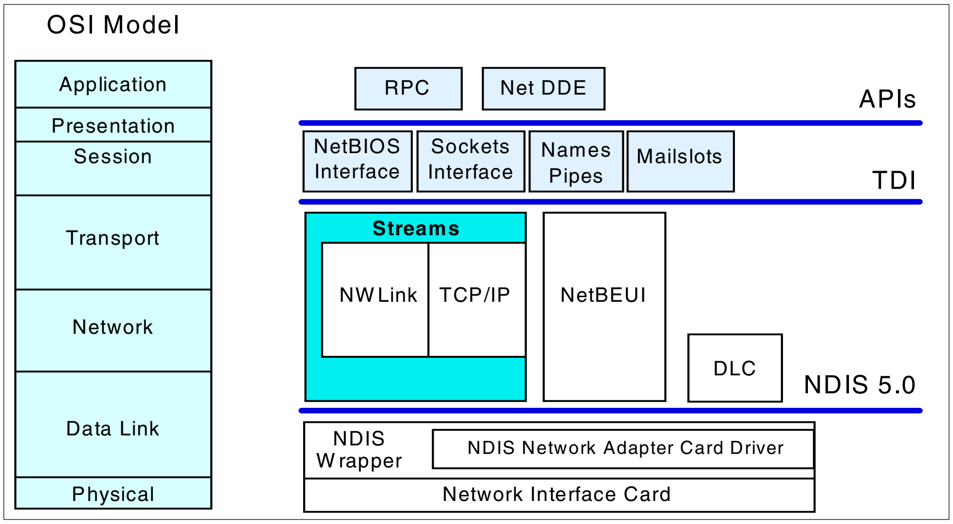


Figure 144. Windows Network Programming Interface

The following sections briefly describe these mechanisms.

10.3.13.1 NetBIOS

The NetBIOS interface provides the NetBIOS mapping layer between NetBIOS applications and the TDI protocol. A NetBIOS client/server application can communicate over NetBEUI, NWLink, and NetBIOS over TCP/IP. Because NetBIOS has been used as a network API since the early 1980s, this interface is available in Windows 2000 for compatibility reasons.

10.3.13.2 Windows Sockets

Sockets are a standard networking mechanism originally developed at the University of California, Berkeley. Windows Sockets is an extension of sockets for the Windows environment. Windows 2000 implements a 32-bit version of Windows Sockets. The Windows Sockets interface in Windows 2000 supports both NWLink and TCP/IP transport protocols. Windows 2000 also accepts standard Windows Sockets calls from applications and translates them into equivalent TDI calls.

10.3.13.3 Named Pipes

Named Pipes provides messaging services allowing applications to share memory over the network. These services are connection-oriented, which means that the sender and receiver require acknowledgment of successful and unsuccessful receipts.

Named Pipes are used when reliability is required over the network. They are implemented as a file system in Windows 2000.

10.3.13.4 Mailslots

Mailslots provide connectionless messaging services on a Local Area Network (no acknowledgment of successful or unsuccessful receipts is required). Windows 2000 uses mailslots for the registration of computers, workgroups, or domains, user names on the network, for the Browser service, and for sending broadcast messages to computers or users.

As with the Named Pipes, mailslots are implemented in Windows 2000 as file systems. As such, remote access to named pipes and mailslots is done through the redirector.

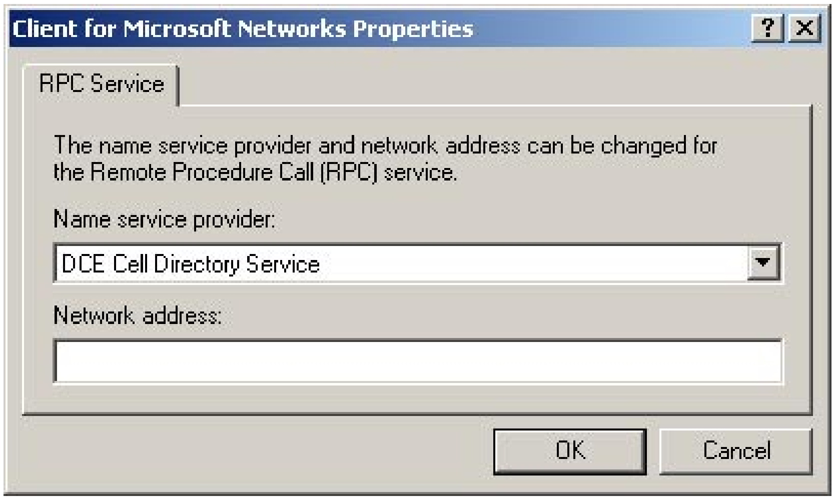
10.3.13.5 NetDDE

Network Dynamic Data Exchange (NetDDE) is a network extension of Dynamic Data Exchange (DDE). DDE allows one application to communicate with another application and exchange and share data. NetDDE allows communication between applications across the network. NetDDE uses the NetBIOS API to communicate with the underlying network components.

10.3.14 DCE for Windows 2000

Windows 2000 is designed to integrate with DCE at the RPC level. The Microsoft RPC implementation is quickly becoming feature-complete, and provides strong protocol level compatibility with other DCE implementations.

Windows 2000 comes with a DCE Cell Directory Service that could be configured as the name service provider for your Client for Microsoft Network service, shown in Figure 145 on page 484.



The name service provider and network address can be changed for the Remote Procedure Call (RPC) service.

Name service provider: DCE Cell Directory Service

Network address:

Figure 145. DCE Cell Directory Service

10.3.15 RAS

Remote Access Service (RAS) is a Windows 2000 service that allows remote workstations to access Windows 2000 RAS servers across standard telephone lines, X.25 packet switched networks or ISDN lines. Once connected to the RAS server, the user can access resources from the network to which the RAS server is connected as if the user was working at the office. The RAS server acts as a gateway between the RAS client and the network.

Microsoft RAS is designed to work with per-user information stored in the domain controller or on a RADIUS server. Using a domain controller simplifies system administration because dial-up permissions are a subset of the per-user information that the administrator is already managing in a single database.

Microsoft RAS was originally designed as an access server for dial-up users. RAS is also a tunnel server for PPTP and L2TP connections. Consequently, these Layer 2 VPN solutions inherit all of the management infrastructure already in place for dial-up networking.

In Windows 2000, RAS takes advantage of the new Active Directory, an enterprise-wide, replicated database based on the Lightweight Directory Access Protocol (LDAP). LDAP is an industry-standard protocol for accessing directory services and was developed as a simpler alternative to the X.500 DAP protocol. LDAP is extensible, vendor-independent, and standards-based. This integration with the Active Directory allows an administrator to assign a variety of connection properties for dial-up or VPN sessions to individual users or groups. These properties can define per-user filters, required authentication or encryption methods, time-of-day limitations, and so on.

10.3.16 Interoperability with other platforms

The following sections discuss interoperability with various other platforms.

10.3.16.1 Interoperability with Netware

Included with both Windows 2000 Servers and Windows 2000 Professional, NWLink supports connectivity between computers running Windows 2000 and those running NetWare systems. IPX support preserves investments in legacy NetWare networks by making it easy to integrate them with Windows 2000 Server.

Included with Windows 2000 Server, Gateway Services for Netware (GSNW) enables a computer running Windows 2000 Server to connect to computers running NetWare 3.x or 4.x server software. Logon script support is also included. In addition, administrators can use GSNW to create gateways to NetWare resources, allowing computers running only Microsoft client software to gain access to NetWare resources.

A separate Microsoft product, Services for Netware (SFN), enables a computer running Windows 2000 Server to provide file and print services directly to NetWare 5.x and compatible client computers. The server appears just like any other NetWare server to the NetWare clients, and the clients can gain access to volumes, files, and printers at the server. No changes or additions to the NetWare client software are necessary.

10.3.16.2 Interoperability in a UNIX environment

Microsoft offers an optional software called Services for UNIX version 2.0 (SFU) that delivers both NFS Server and NFS Client software as well as password synchronization and NIS integration. The NFS client allows clients to gain access to files that exist on UNIX servers, and the NFS server allows UNIX workstations and servers to gain access to files on systems running Windows 2000.

Running the SMB protocol on UNIX servers and workstations, UNIX systems can gain access to files managed by Windows 2000 Server. However, using Windows-based clients to gain access to files held on UNIX servers is more common. Windows-oriented networking on UNIX includes a freeware product known as SAMBA and AIX FastConnect on IBM @serverpSeries. For a more in depth discussion on Windows 2000 and AIX interoperability, see the Redbook *AIX 5L and Windows 2000: Solutions for Interoperability,* SG24-6225-00, downloadable from:

`http://www.redbooks.ibm.com.`

10.3.16.3 Interoperability with Macintosh

Services for Macintosh lets Intel-based and Apple Macintosh clients share files and printers and remotely connect to a Microsoft network.

File Server for Macintosh (MacFile) lets administrators designate a directory as a Macintosh-accessible volume and ensures Macintosh filenames are legal NTFS names and handles permissions.

Print Server for Macintosh (MacPrint) lets Apple users send print jobs to a spooler on the Windows 2000 Server.

10.3.17 Windows 2000 networking model

Windows 2000 supports two networking models: The Workgroup model and the Domain model. The Windows 2000 domain model has been completely redesigned since the previously available version in Windows NT. In this section, we are going to deal with the Windows 2000 networking model providing an introduction to its architecture and functions.

10.3.17.1 Workgroup model

A workgroup is simply a collection of computers that are grouped together. These computers can share files or printers in a peer-to-peer relationship. Each computer can share its own resources, but no computer is the sole resource master. There is no centralized user account administration and instead, each computer maintains its own user account and security database. A stand-alone system is defined as a workgroup consisting of only one computer. Similar to UNIX, this model is a collection of stand-alone systems without NIS or DCE and with NFS to get access to shared file systems via the automount facility.

10.3.17.2 Domain model

A domain is a collection of computers that share a common user account database and common security policy. This greatly facilitates the administration of all the systems. In a domain, the system administrator needs to create one account per user for the entire domain. Whatever the physical computer the user wants to work on, he or she will only be able to log into the computer if this computer is part of the domain.

When a user logs on to a domain, he or she gains access to all resources shared within the domain. These resources can be located on any server in the domain but to the user, it appears as though they are connected to a single server.

Similar to UNIX, this model is similar to a DCE cell and to the DFS filespace concept.

10.3.18 Active Directory

A directory is an information source used to store information about objects. In a distributed computing system or a public computer network, such as the Internet, there are many objects, such as printers, fax servers, applications, databases, and other users. Users want to find and use these objects. Administrators want to manage how these objects are used.

The Active Directory is the directory service included with Windows 2000 Server. It extends the features of previous Windows-based directory services and adds entirely new features. The Active Directory is secure, distributed, partitioned, and replicated. It is designed to work well in any size installation, from a single server with a few hundred objects to thousands of servers and millions of objects.

The Active Directory is not a true X.500 directory. Instead, it uses LDAP as the access protocol and supports the X.500 information model without requiring systems to host the entire X.500 overhead.

10.3.18.1 Terms and definitions

Here, we are going to introduce the main terms and definitions that are typical of an Active Directory.

name space

As is any directory service, the Active Directory is, primarily, a name space. A telephone directory is a name space. A name space is any bounded area in which a given name can be resolved. Name resolution is the process of translating a name into some object or information that the name represents. The UNIX file system forms a name space in which the name of a file can be resolved to the file itself.

The Active Directory forms a name space in which the name of an object in the directory can be resolved to the object itself.

Object

An object is a distinct named set of attributes that represents something concrete, such as a user, printer, or application. The attributes hold data describing the subject that is identified by the directory object. Attributes of a user might include the user's given name, surname, and e-mail address.

Domains

A domain is a single security boundary of a Windows NT or Windows 2000 computer network. The Active Directory is made up of one or more domains. On a stand-alone workstation, the domain is the computer itself. A domain can span more than one physical location. Every domain has its own security policies and security relationships with other domains.

Domain Trees

A domain tree (a tree) is comprised of several domains that share a common schema and configuration forming a contiguous name space. Domains in a tree are also linked by trust relationships. The Active Directory is a set of one or more trees. A Windows 2000 domain tree is a hierarchy of Windows 2000 domains, each consisting of a partition of the Active Directory. The shape of the tree and the relationship of the tree members to each other are determined by the DNS names of the domains. The domains in a tree must form a contiguous name space so that a.myco.com is a child of myco.com, b.myco.com is a child of a.myco.com, and so forth.

The Windows 2000 domain tree is the enterprise-wide Active Directory. All Windows 2000 domains in a given enterprise should belong to the enterprise domain tree. Enterprises that need to support disjointed DNS names for their domains will need to form a forest.

Figure 146 on page 489 shows an example of a domain tree. Looking at this example, you may note how the domain tree is strictly related to DNS name space.

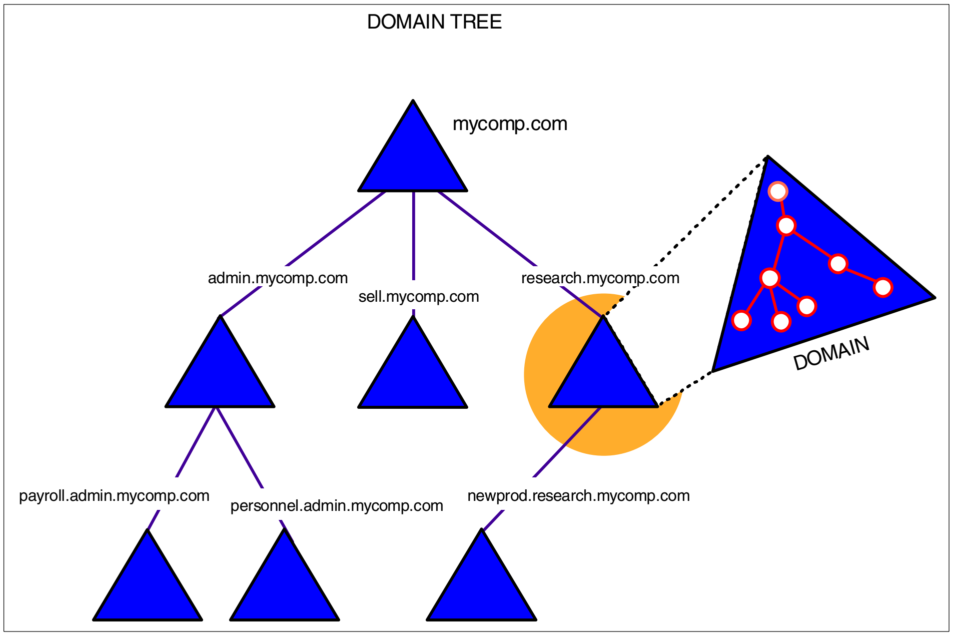


Figure 146. A domain tree

Forests

A forest is a set of one or more trees that do not form a contiguous name space. All trees in a forest share a common schema, configuration, and Global Catalog. All trees in a given forest trust each other via transitive hierarchical Kerberos trust relationships.

Figure 147 on page 490 shows an example of a forest domain. Note how the forest is composed of disjoined trees and how the DNS name space in the example is disjoined as well.

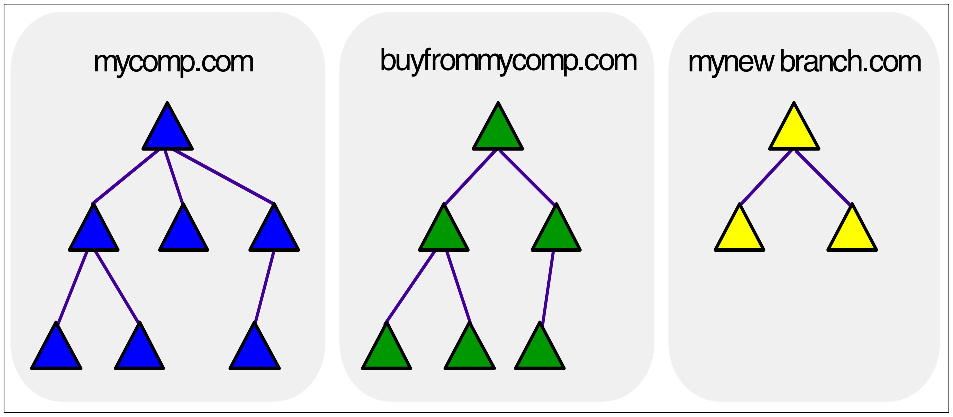


Figure 147. Domain forest

10.3.18.2 Active Directory features

This section describes some of the major features and components of the Active Directory.

DNS Integration

The Active Directory is tightly integrated with the Domain Name System (DNS). DNS is the distributed name space used on the Internet to resolve computer and service names to TCP/IP addresses. Most enterprises with intranets use DNS as the name resolution service. The Active Directory uses DNS as the location service.

Windows 2000 domain names are DNS domain names. For example, "mycompany.com" is a valid DNS domain name and could also be the name of a Windows 2000 domain.

Active Directory servers are published via Service Resource Records (SRV RRs) in DNS. The SRV RR is a DNS record used to map the name of a service to the address of a server offering that service. The name of a SRV RR is in this form:

```
<service>.<protocol>.<domain>
```

Active Directory servers offer the LDAP service over the TCP protocol so that published names are in the form:

```
ldap.tcp.<domain>
```

Supported standard protocols

The Active Directory supports the following standard protocols:

- **LDAP** - The Active Directory core protocol is the Lightweight Directory Access Protocol (LDAP). LDAP Version 2 and Version 3 are supported.
- **MAPI-RPC** - The Active Directory supports the remote procedure call (RPC) interfaces supporting the MAPI interfaces.
- **X.500** - The Active Directory information model is derived from the X.500 information model.

Application Programming Interfaces (APIs)

The Active Directory supports the following APIs:

- **ADSI** - Active Directory Service Interfaces (ADSI) provide a simple, powerful, object-oriented interface to the Active Directory. Developers can use many different programming languages, including Java, the Visual Basic programming system, C, C++, and others. ADSI is fully scriptable for ease of use by system administrators. ADSI hides the details of LDAP communications from users.
- **LDAP API** - The LDAP C API, defined in RFC 1823, is a lower-level interface available to C programmers.
- **MAPI** - The Active Directory supports MAPI for backward compatibility. New applications should use ADSI or the LDAP C API.

10.3.18.3 Security

This is only an overview of security in the Active Directory. For more information about the Windows 2000 security model, refer to section 7.3, “Windows 2000 security” on page 187. The main aspects related to security in Windows 2000 are:

- **Object Protection** - All objects in the Active Directory are protected by Access Control Lists (ACLs). ACLs determine who can see the object and what actions each user can perform on the object. The existence of an object is never revealed to a user who is not allowed to see it.

  An ACL is a list of Access Control Entries (ACEs) stored with the object it protects. In Windows 2000, an ACL is stored as a binary value called a Security Descriptor. Each ACE contains a Security Identifier (SID), which identifies the principal (user or group) to whom the ACE applies and information on what type of access the ACE grants or denies.
- **Delegation** - Delegation is one of the most important security features of the Active Directory. Delegation allows a higher administrative authority to grant specific administrative rights for containers and subtrees to individuals and groups. This eliminates the need for *domain administrators* with sweeping authority over large segments of the user population.

ACEs can grant specific administrative rights on the objects in a container to a user or group. Rights are granted for specific operations on specific object classes via ACEs in the container's ACL.

- **Inheritance** - Inheritance lets a given ACE propagate from the container where it was applied to all children of the container. Inheritance can be combined with delegation to grant administrative rights to a whole subtree of the directory in a single operation.
- **Private Key Security** - Along with the Active Directory, the next release of Windows 2000 Server will implement a distributed security model. This distributed security model is based on the MIT Kerberos authentication protocol. Kerberos authentication is used for distributed security within a tree and accommodates both public and private key security using the same Access Control List (ACL) support model of the underlying Windows 2000 operating system. The Active Directory replaces the registry account database and is a trusted component within the Local Security Authority (LSA).

  A single sign-on to the Windows 2000 domain tree allows user access to resources anywhere in the corporate network.
- **Public Key Security** - The Active Directory also supports the use of X.509 v3 Public Key Certificates for granting access to resources for subjects (for example, users) that do not have Kerberos credentials. This type of user is most often someone from outside an organization who needs access to resources within the organization.

10.3.18.4 Domain Controllers

Domain Controllers keep a copy of the directory. In Windows NT 3.51 and 4.0 domains, there are two kinds of Domain Controllers: Primary Domain Controllers (PDCs) and Backup Domain Controllers (BDCs). A domain must have exactly one PDC but can have virtually any number of BDCs. Primary Domain Controllers hold a read/write copy while Backup Domain Controllers hold a read-only copy. Hence, all updates must be made on the PDC.

In Windows 2000, all Domain Controllers for a given domain hold a writable copy of the directory. There is no distinction between primary and backup; all domain controllers are equal. For this reason, administration is further simplified because there is no notion of a primary domain controller (PDC) or backup domain controller (BDC) as in previous NT versions. The Active Directory only uses domain controllers (DCs), and all DCs are peers. An administrator can make changes to any DC, and the updates will be replicated on all other DCs.

The Active Directory provides multi-master replication. Multi-master replication means that all replicas of a given partition are writable. This allows updates to be applied to any replica of a given partition. The Active Directory replication system propagates the changes from a given replica to all other replicas. Replication is automatic and transparent.

The multi-master synchronization mechanism relies on Update Sequence Numbers to maintain up-to-dated all the replicas. Some directory services use timestamps to detect and propagate changes. In these systems, it is very important to keep the clocks on all synchronized directory servers. Time synchronization in a network is very difficult. Even with very good network time synchronization, it is possible for the time at a given directory server to be incorrectly set. This can lead to lost updates.

Windows 2000 provides distributed time synchronization, but the Active Directory replication system does not depend on time for update propagation. Instead, the Active Directory replication system uses Update Sequence Numbers (USNs). A USN is a 64-bit number maintained by each Active Directory server. When the server writes any property to the Active Directory, the USN is advanced and stored with the property written. This operation is performed automatically.

10.3.18.5 Global catalog

The Active Directory can consist of many partitions or naming contexts. The distinguished name (DN) of an object includes enough information to locate a replica of the partition that holds the object. Many times, however, the user or application does not know the DN of the target object or which partition might contain the object. The Global Catalog (GC) enables users and applications to find objects in an Active Directory domain tree if the user or application knows one or more attributes of the target object.

The Global Catalog contains a partial replica of every user-naming context in the directory. It contains the schema and configuration naming contexts as well.

10.3.18.6 Scalability

The Active Directory scales by creating one copy of the directory store for each domain. This copy of the directory store holds the objects that apply to that domain only. If multiple domains are related, they can be built into a tree. Within this tree, each domain has its own copy of the directory store, with its own objects and the ability to find all the other copies in the directory store tree.

Rather than creating a single copy of the directory that gets larger and larger, the Active Directory creates a tree made up of small pieces of the directory, each containing information that allows it to find all the other pieces. The Active Directory breaks the directory into pieces so that the most-often-used part of the directory is closest to them. Other users in other locations may want to use that same part of the directory, and they would also have a copy close to them. All replicas of that part of the directory are kept synchronized. If a record in any copy is modified, the change is propagated to the other copy. This allows the Active Directory to scale up to many millions of users in a tree.

Using domain trees and forests

The key to the scalability of Active Directory is the domain tree. Unlike directory services that consist of a single tree structure and require a complex top-down partitioning process, the Active Directory provides a simple and intuitive bottom-up method for building a large tree. In the Active Directory, a single domain is a complete partition of the directory.

A single domain can start very small and grow to contain over 10 million objects. When a more complex organizational structure is required or a very large number of objects must be stored, multiple Windows 2000 domains can be easily joined together to form a tree. Multiple trees can form a forest. Forests allow an enterprise to include Windows 2000 Domains with disjoint DNS names, for example, MyCo.com and MyCoResearch.com.

10.3.18.7 Backward compatibility

A critical need for customers who have installed Windows NT Server versions 3.51 or 4.0 is backward compatibility. From the start, the Active Directory was designed with backward compatibility built-in. The Active Directory provides complete emulation of the Windows NT domain model and authentication requirements.

Chapter 11. Scalability and high availability

The importance of both hardware and software scalability has increased dramatically over the past decade or so as businesses have placed more dependence on their computing systems. The same can be said of the need for high availability of hardware and software. While hardware improvements have been moving ahead at a fast pace, software, specifically, operating systems, have needed to be able to deal with increased processor speed and memory while ensuring that performance is always optimum.

With companies relying so heavily on their systems and expanding, the ability of the systems to grow with the business and remain at an acceptable performance level has never been more important.

11.1 Scalability

This section looks at the increasing need for operating systems to be able to perform with an increased number of processors and memory while utilizing these resources in the best possible way.

11.1.1 AIX scalability

This section gives an overview of AIX scalability.

11.1.1.1 SMP support

AIX Version 5L can run on anything from a standalone graphics workstation used for CAD/CAM applications to a 512-node SP (scalable parallel) system used to gather geophysical details or defeat a world chess champion.

On SMP systems, AIX has been extremely well tuned and optimized to exploit the pSeries SMP architecture, and, with the release of AIX Version 4.3.3, the introduction of the Workload Manager provided a tool for managing the system workload and resources. We will cover AIX Workload Manager in more detail in section 11.1.1.3, “AIX Workload Manager (WLM)” on page 498.

AIX 4.3.3 boasted superior kernel scalability enhancements. These enhancements nearly tripled online transaction processing throughput while only requiring a doubling of the number of CPUs (12 to 24), memory (32 GB to 64 GB), and increasing processor speed. This increased throughput is especially meaningful in Enterprise Resource Planning (ERP) and OnLine Transaction Processing (OLTP) applications. Individual results may vary due to different applications and workload combinations.

Memory management was improved in AIX 4.3.3 to allow higher concurrence with multiple frame lists and multiple page replacement daemons. This reduces contention in the serialization mechanisms and allows processes with lower latencies to service the memory requests.

AIX 5L organizes the runnable threads into per-cpu local run queues. This simplifies the process of determining which thread to run next and eliminates the costly logic that was necessary to promote good cache affinity when scheduling from a single global run queue. It also virtually eliminates locking contention in this performance-critical subsystem.

With local run queues, the dispatchers' affinity algorithms have been strengthened resulting in greatly improved throughput on busy SMP systems. User mode threads generate less cache interference as a result of their increased affinity. These reductions in system overhead translate directly into increased application throughput.

Figure 148 on page 497 shows the Relative Online Transaction Processing (ROLTP) comparisons between the IBM @server pSeries PCI enterprise server range. ROLTP is an estimate of commercial processing performance derived from an IBM analytical model that simulates some of the system's operations, such as CPU, cache, and memory. Figure 148 on page 497 also shows the improvement in performance, by machine, as the number of processors increases. In particular, the last four rows of the graph show the model S85 differing in ROLTP with 6, 12, 18, and 24 processors.

Similarly, Figure 149 on page 498 shows the comparison of ROLTP for SP nodes.

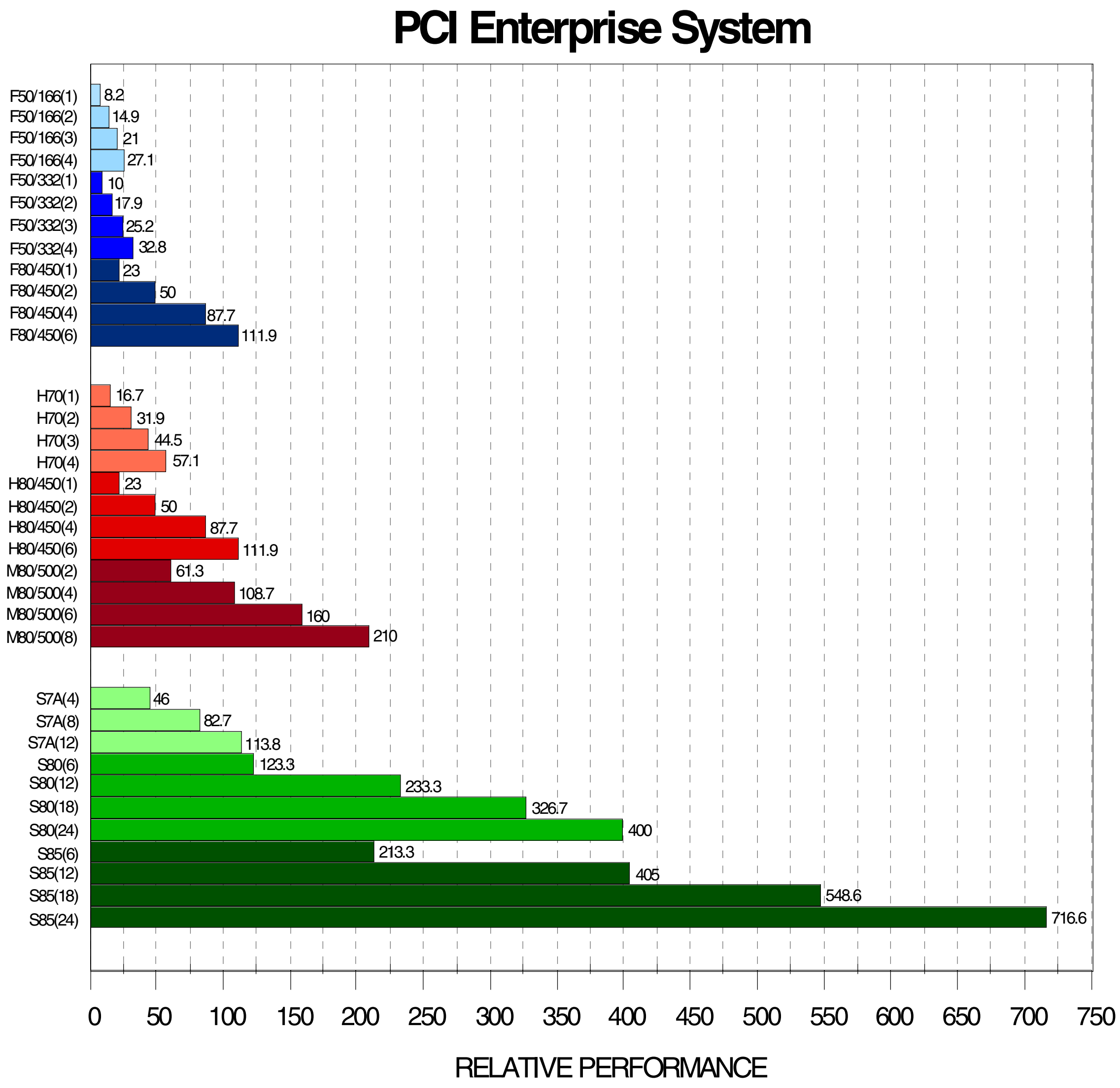


Figure 148. ROLTP of the IBM eServer pSeries PCI Enterprise Systems

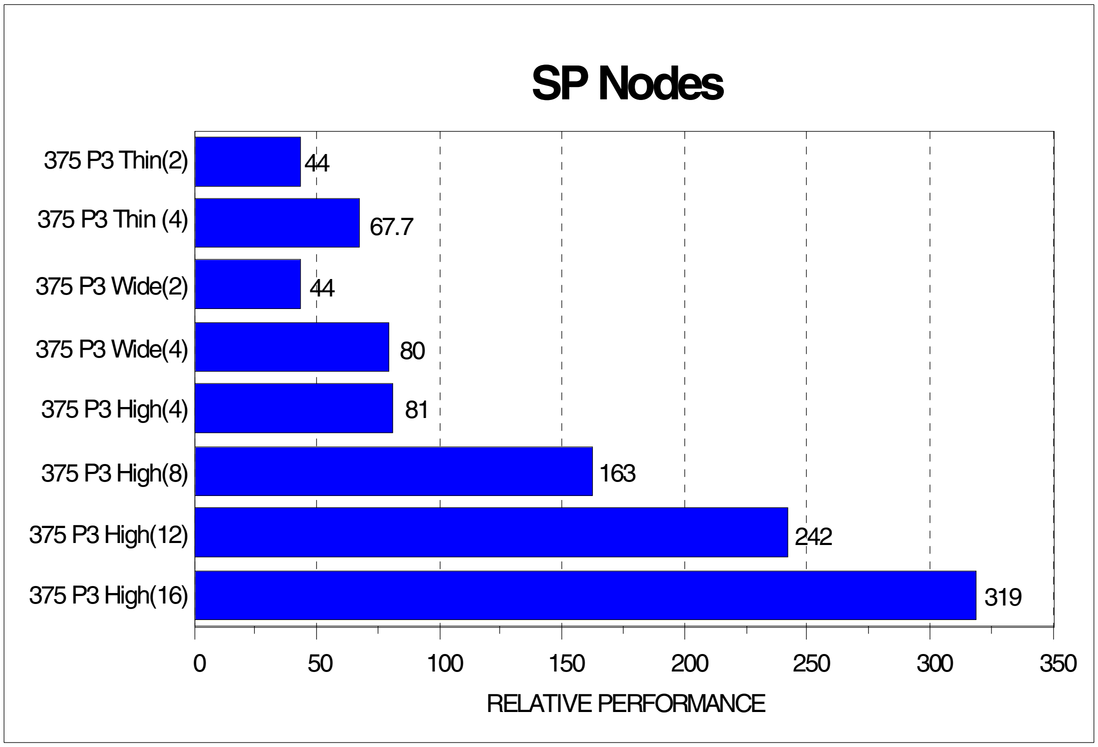


Figure 149. ROLTP of the IBM eServer pSeries SP nodes

11.1.1.2 Capacity Upgrade on Demand (CUoD)

A new offering with the 7017-S85, or @serverpSeries 680, CUoD allows a customer to order a 680 with 6, 12, or 18 processors and specify another 6-way processor block which is inactive. If usage increases to a point where the customer requires more cpu power they can call into IBM to have the extra 6-way processor block enabled.

This offering allows customers more flexibility in deciding how far they wish their 680 system to scale, and offers extra processing capacity in times of critical need.

11.1.1.3 AIX Workload Manager (WLM)

The AIX Workload Manager system introduced with AIX 4.3.3 provides a policy-based method for managing system workload and system resources. AIX Workload Management includes the following capabilities:

- It defines system resource allocations that can be applied towards specific jobs or job classes. The operating system allocates CPU and memory resources to jobs or job classes in accordance with defined resource allocation policies.

- It helps ensure critical applications are not impacted by less important jobs in the system during peak demand.
- It allows logical job separation on the server.
- It permits applications to remain in memory for more predictable performance.
- It helps provide greater convenience and control by using both resource targets and resource limits.
- It allows policies to be set by the system administrator once with no further interaction required. The system will automatically apply the specified policies and adjust for changing conditions.
- It permits the creation and management of 29 classes of jobs, each with different resource policies and system administrator-specified names.
- It allows the creation of automatic classification rules to assign processes to classes.
- It permits usage of nine tiers of jobs where each tier's resource needs are satisfied before resources are provided to jobs in the next tier.
- Provides control options that include minimum and maximum percentage limits, shares, or a combination of both.
- Management of disk I/O bandwidth, in addition to the already existing CPUcycles and real memory.
- Graphic display of resource utilization.
- Performance Toolbox integration with WLM classes, enabling the toolbox to display performance statistics.
- Fully dynamic configuration, including setting up new classes without restarting WLM.
- Application Programming Interface (API) to enable external applications to modify the system’s behavior.
- Manual reclassification of processes, which provides the ability to have multiple instances of the same application in different classes.
- More application isolation and control:
  - New subclasses add ten times the granularity of control (from 27 to 270 controllable classes).
  - Administrators can delegate subclass management to others users and groups rather than root or system.

- Possibility of inheritance of classification from parent to child processes.

- Application path name wildcard flexibility extended to user name and group name.
- Tier separation enforced for all resources, enabling a deeper prioritization of applications.

These capabilities can be easily managed through Web-based System Manager, SMIT, shell scripts, or command line interfaces. Web-based System Manager enables management of AIX systems on the Internet from anywhere via an intuitive, object-oriented, easy-to-use GUI.

11.1.1.4 Parallel systems support

AIX also supports the IBM @server pSeries SP (Scalable POWER parallel). The IBM @server pSeries SP is a massively parallel system that uses IBM @server pSeries systems as nodes. Up to 512 nodes can be installed in a given SP system. Each node is an IBM @server pSeries that can be a uniprocessor system or an SMP system (two to eight-way). All the nodes can be interconnected to other nodes via a high-speed switch (150 MB/s) that allows any node to communicate with any other node. The SP brings scalable parallel performance to the top of the IBM @server pSeries server range. It delivers a major leap forward in cost-effective, high-performance, parallel computing that was once considered too expensive or too complex to pursue.

The SP is a shared-nothing system architecture, meaning that memory is not shared (distributed memory); I/O resources are not physically shared between the processors, and each node runs its own operating system (AIX). This type of architecture allows an extremely high scalability since there is no hardware resource contention due to sharing of memory, I/O, and so on.

The scalable capabilities of the SP system allow customers to scale their applications, both in computation and data, beyond what is possible with conventional uniprocessor systems or SMP systems.

From a system management perspective, the SP can be considered one system. The Parallel System Support Program (PSSP) that comes with the SP systems helps manage the SP as one system and hides the complexity of a high number of individual systems. SP system management is performed from a single place, the control workstation.

The SP is particularly well-suited for database query, decision support, and business management.

The SP initial focus was on high-performance scientific and technical computing in areas, such as computational chemistry, petroleum exploration and production, engineering analysis, research, and “grand challenge” problems (those important to the national interest). Today, SP systems address those areas and are also being used for commercial computing – primarily complex query, decision support, data mining, data warehousing, business-management applications, and online transaction processing. It has also been extensively used for LAN consolidation.

11.1.1.5 Remote System Management

A significant consideration in an enterprise environment with numerous systems is administration of systems. With AIX Version 5L, it is equally easy to administer local and remote systems, even over low-speed modem lines or networks. This is because AIX includes all TCP/IP daemons (telnetd, rlogind) and utilities, such as ATE (Asynchronous Terminal Emulation), BNU, and so on, which allow the administrator to connect to any system and access the command line interface of the remote system as well as the ASCII version of SMIT. It is also possible to monitor the system remotely with the Web-based System Manager.

Once connected to the remote system, many tasks can be performed, such as software update, fix installation, diagnostics online, reboot of the system, and user/group creation. In fact, all tasks that do not absolutely require physical access to the system can be performed remotely.

This is a standard facility in AIX. It is also possible to use system-management products, such as TME 10, to perform more complex tasks, such as software distribution, user administration, security management, systems monitoring, network management, and so on.

11.1.2 Windows 2000 scalability

This section gives an overview of Windows 2000 scalability.

11.1.2.1 SMP support

The four packages in Windows 2000 are designed to meet different business needs, and, therefore, they support different numbers of processors. Table 27 on page 502 shows the number of processors and physical memory that each Windows 2000 package supports:

Table 27. Maximum processors and memory in each Windows 2000 package

Windows 2000 Package	Max. Processors	Max. Memory
Professional	2	4GB
Server	4	4GB
Advanced	8	8GB
Datacenter	32	64GB

11.1.2.2 Web Server scalability

With the increased Windows 2000 SMP support, the scalability of its Web server has also improved. According to Microsoft, the most important Web technology that is integrated with the Windows 2000 Server package is the Internet Information Services 5.0 (IIS). This is designed to allow users to share information as well as host and manage sites on the Web. It is meant to deliver higher levels of Web server uptime as well as provide advanced crash protection for the Web applications based on the server. It also takes advantage of up to eight-way SMP by scaling to the large workloads on the Internet.

Another feature of Windows 2000 Server and Web server scalability is the ability to host multiple sites on a single IP address.

The only drawback to the scalability of the Windows 2000 range is that they are separate packages. Should a machine need to grow outside of what the current package supports, an upgrade to the next level of Windows 2000 will also be necessary in order to support the processing power of the upgraded/new machine. This will become a limitation for rapidly growing businesses unless pre planning is done, growth anticipated, and a larger machine and licence are purchased up front.

11.1.2.3 Remote system management

There have been many updates to Windows 2000, especially the remote management tools that have been made available. New to the Windows 2000 Server is the ability to:

- Remote management using Terminal Services.
- Remote management using the Administrative tools and MMC.
- Script with Windows Scripting Host (WSH).
- Perform remote installation of Windows 2000 Professional and applications.

Terminal services

Terminal services allow remote administration of the Windows 2000 Server using an integrated terminal emulation service in Windows 2000 Server allowing clients to access Windows-based applications running on the server. More detailed information about Terminal Services is available in section 8.2.9, “Windows 2000 Terminal services” on page 325.

Administrative Tools

The Administrative Tools comes with the Windows 2000 Server CD and is a set of snap-ins available in the MMC. They are automatically installed on servers and have to be manually added to WIndows 2000 professional installation by running the adminpak.msi file from the %SystemRoot\System32% directory. Using this tool, it is possible to manage any Windows 2000 server remotely from any computer also running Windows 2000.

Windows Scripting Host (WSH)

Using WSH allows automation of actions, such as the creation of shortcuts or connecting and disconnecting from a network server. This can be achieved by either clicking on the shortcut icon or by typing the name of the script at the command prompt. There are windows-based and command-based versions, and either can be run from the command line by typing either `wscript.exe` or `cscript.exe`. The languages it supports are Visual Basic Scripting Edition (VBScript), JScript or Perl but can easily be extended to support other scripting languages.

Remote Installation Services

The Remote Installation Services (RIS) allow Windows 2000 Professional to be remotely installed on any client computer that can be started remotely.

11.2 High availability

Increasingly, enterprise systems are relied upon to provide services that are mission-critical to organizations. Applications and databases are outgrowing even the largest single systems and introducing new complexities and costs to data management, and companies are looking for greater flexibility in system configuration and upgrades to deal with rapid workload changes. Until recently, most organizations overcame these challenges by using SMP machines, but, as processing requirements continue to grow, many are finding that they must move towards clusters.

The availability of data and applications is of paramount importance. Planned and unplanned outages need to be minimized or even eliminated. Unplanned outages are caused by hardware failure, software errors, operator errors, and environmental problems. Planned outages are availability interruptions required during hardware or software administration. These outages are further reduced by building in redundancy. Selecting the right method of improving availability is influenced by the business impact of downtime and hours of operation. Two methods that are used to improve availability are Fault Tolerance and High Availability.

Fault tolerance relies on specialized hardware to detect a hardware fault and instantaneously switch to a redundant hardware component. Since the component is not actually used until that time, this is a very expensive model to use. This model generally does not address software failures either, and this can be the most common reason for down-time.

High Availability views availability as a set of system-wide shared resources that cooperate to guarantee essential services. This eliminates a single point of failure. A popular method of high availability is clustering. This is a group of independent servers networked together to share critical resources. By clustering two or more servers to back up critical applications, a business can utilize more of its resources; this is a definite advantage over the fault tolerance model.

With the improvements that have been made to hardware, it is not a major reason for computer system unavailability. Instead, application software failure, software maintenance, network failure, and planned upgrades are the major sources of downtime.

High availability is a combination of many components including:

- Hardware
- Network
- Operating System
- Applications
- Clustering

A good combination of all components enables systems to be highly available. A failure in any single component can compromise the overall availability of the system.

11.2.1 AIX high availability

IBM provides two high availability solutions that run on AIX and are based on the clustering concept. High Availability Cluster Multiprocessing (HACMP) for

AIX Version 4.3 is a control application that, when using the Enhanced Scalability feature, can link up to 32 IBM @server pSeries servers or SP nodes into highly available clusters. Clustering servers or nodes enables parallel access to their data, which can help provide the redundancy and fault resilience required for business-critical applications. Clustering also offers gradual, scalable growth. Upgrading a machine or just adding a processor is a simple task. HACMP includes graphical user interface-based tools to help install, configure, and manage your clusters in a highly-productive manner.

The other product is designed to work over a wide geographic area and enhances HACMP. This product is called High Availability Geographic Cluster (HAGEO).

11.2.1.1 HACMP Version 4.3 overview

HACMP is flexible in configuration and use. Uniprocessors, SMPs, and SP nodes can all participate in highly-available clusters. You can mix and match system sizes and performance levels as well as network adapters and disk subsystems to satisfy your application, network, and disk performance needs. It is also simple to use because installation and management can be done from a single console.

Planned downtime can account for a large part of total system downtime. HACMP allows planned downtime to be minimized by concurrently performing hardware, software, and other maintenance activity while applications continue to run on other nodes. The Dynamic Reconfiguration utility allows cluster resources to be added or removed without cluster disruption, enabling continuous operation in the light of maintenance. Service may be moved from one cluster node to another and then returned after the maintenance activity is complete.

Unplanned downtime can have one of two causes: Hardware or software. Together with facilities provided by the AIX operating system, HACMP can protect your operation from hardware failures by automatically moving services from a failing node to other cluster nodes. Figure 150 on page 506 is an example of a disk takeover when a machine fails.

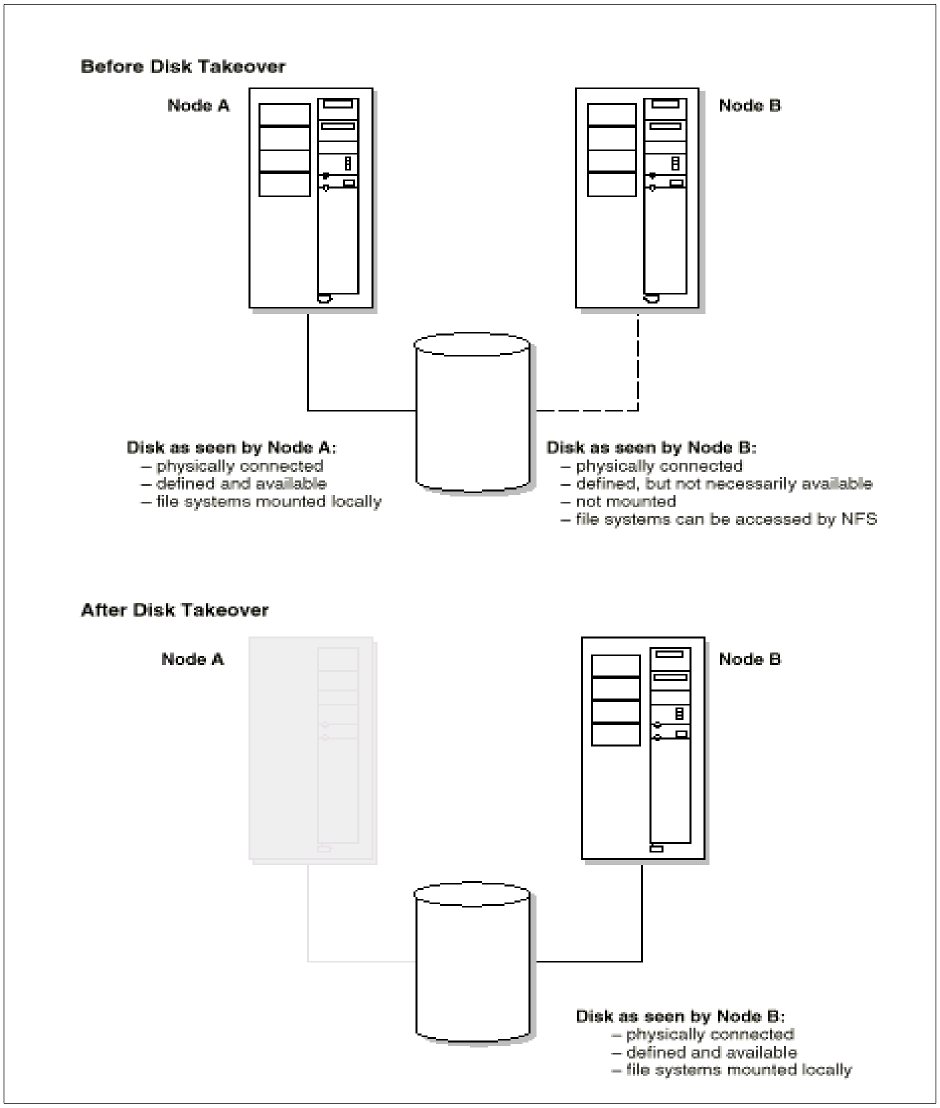


Figure 150. HACMP disk takeover

The HACMP Cluster Manager monitors the processors and the network interfaces using a heartbeat protocol. When a configurable number of failed Keepalive Packets is reached, the Cluster Manager assumes failure and takes action. Failure is not recognized until all active Cluster Managers agree that a failure has occurred. It is essential that all processors within the cluster agree the reason for failure. This is why IBM HACMP does not rely on a single route to carry the heartbeats between cluster processors. Heartbeats (keepalive packets) operate across TCP/IP adapters, RS232 serial link, and SCSI twin tailed chain. For example, should the TCP/IP software stack fail, this could be viewed as a complete node failure if the heartbeat only operates across TCP/IP adapters.

Figure 151 demonstrates how HACMP monitors the processors with keepalive packets between different processors (nodes) on the cluster:

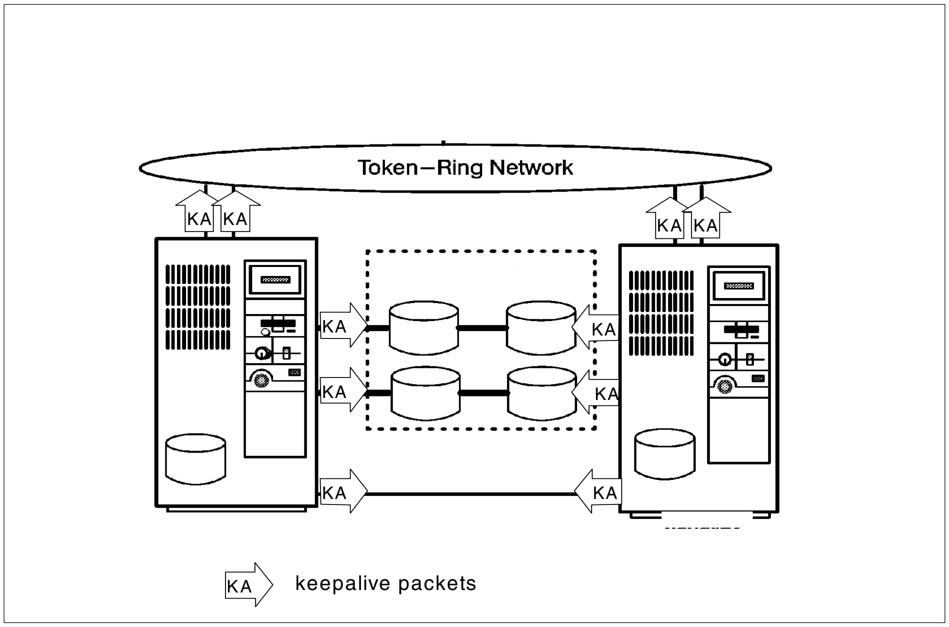


Figure 151. HACMP event detection

HACMP also provides an API to allow client applications to communicate with the Cluster Manager services. Other features of HACMP include Cluster Quick Configuration, Cluster Node Snapshot, and Failure Emulation. Cluster Quick Configuration provides predefined cluster configuration options for automatic cluster configuration. It is also possible to add customized configurations to this. Cluster Node Snapshot tracks configuration and facilitates the cloning of additional users. Failure Emulation allows continuous availability by providing emulation utilities to test failover scenarios without actually taking the systems down.

Software failures that cause a node failure can be detected by HACMP, but software failures that interrupt system operation and do not result in a system failure or hang require the next step in availability technology, which is represented by IBM @server pSeries Cluster Technology (RSCT).

Utilizing the Event Management component of RSCT, problems with the operation of software, such as process failures or exhaustion of system resources, can be detected and reacted to before they result in a critical failure. HACMP/ES can monitor, detect, and react to random software failures, thus, allowing your system to remain operational. HACMP/ES can be configured to react to hundreds of system events. In addition to this advanced protection, RSCT allows HACMP/ES to support up to 32 nodes.

HACMP/ES also allows you to define additional cluster events. This allows unrivaled levels of availability using standard components while providing protection against hardware failures and capabilities to perform concurrent maintenance.

HACMP provides a choice of tools for installation, configuration, and system administration tasks: Use either AIX System Management Interface Tool (SMIT) or the Visual System Manager (VSM) GUI.

To help automate the configuration process, you can replicate existing clusters, first by using the information captured by the Cluster Node Snapshot utility and then by using the Quick Configuration utility to replicate the configuration at other sites. With the Dynamic Reconfiguration facility, you can add and remove cluster resources, such as processors, adapters, disk subsystems, and application software, without stopping cluster operations.

In addition, HACMP supports rolling upgrades. This allows the cluster to be upgraded to a new version of HACMP or the operating system without taking your applications down.

With the Cluster Single-Point-of-Control (C-SPOC) facility, common cluster administration tasks can be performed from any node in the cluster, and, with the HAView function, you can use the Tivoli TME 10 NetView for AIX graphical network management interface to monitor clusters and their components across the network from a single node.

The Concurrent Resource Manager of HACMP provides up to eight-way concurrent access to shared disks in a highly-available cluster allowing you to tailor the actions taken during a takeover to suit your business needs.

HACMP works well in conjunction with parallel database products, such as IBM DB2 Universal Database (UDB), to build loosely-coupled parallel clusters that provide high levels of system availability. Data can be split or partitioned on up to 32 nodes for exceptionally scalable cluster performance and high availability.

For NFS environments, HACMP provides the HANFS for AIX feature, which defines two AIX systems as a single highly-available NFS server, eliminating single points of failure. This server can survive hardware and software outages as well as many planned outages that could make a single system NFS server unavailable.

In SP clusters, your data is protected with the highly-regarded Kerberos security protocol from the Massachusetts Institute of Technology/Open Software Foundation (MIT/OSF).

The latest release of HACMP is Version 4.3.1, and this is even more closely integrated with AIX Version 4.3. It has added support for the AIX Fast Connect application as a highly-available resource and an automated AIX error notification facility. This allows automatic detection of what configured resources can be associated with an AIX error log event. The default action is to log these events in the hacmp.out file, but it is possible to create your own scripts to associate with these events.

HACMP cluster configurations

An HACMP cluster can operate with up to eight nodes in numerous configurations. The following examples describe some configuration possibilities using the simplest two-node cluster, but larger numbers of nodes can readily be clustered:

- Idle Standby is the simplest of the HACMP scenarios. It allows for one system to be a backup for another’s resources. The backup is idle, running AIX and the HACMP software but not any applications.
- Simple Fallover is the second of the HACMP scenarios. It allows for one system to act as backup for another’s resources. The primary difference between this scenario and the Idle Standby is that the backup system is running an application. However, the application is not mission-critical and is stopped when a failure of the primary server is detected. It is common for the backup server in this scenario to act as a software development/test platform.
- Mutual Takeover allows each system to run its own mission-critical workload, and each acts as backup to the other. This can cause problems with the sizing of the servers because, in the event of a server failure, the remaining server will have to run both workloads. Solutions to this problem are to:
  - Oversize the servers
  - Accept lower response times after failure
  - Prioritize one workload or set of users after a failure

- Concurrent Access allows multiple servers to provide application service to commonly shared data resource. It is important in this shared environment that data is not damaged inadvertently. Two locking models are supplied to control access to the data.

With clusters of up to eight nodes, the inherent configuration flexibility is tremendous:

- For cost-effective resilience, one node can back up seven nodes.
- Eight nodes can operate during peak operations and drop back to fewer nodes during non-peak hours with minimal interruption of service.
- All resources could go to one node upon failure or be split between several nodes.

A subset of the above modes is Rotating Fallover. Rotating Fallover addresses the problem of reintegrating the failed host once the problem has been rectified.

Instead of the failed host reentering the cluster as the primary server, causing another application outage while users are swapped back, the system simply reenters the cluster as the backup server.

In an HACMP environment, shared disks are physically connected to multiple nodes. SCSI-2 Differential and Serial Storage Architecture (SSA), together with the appropriate cabling, support multiple attachments to external disk subsystems.

There are two main shared disk configurations: Non-concurrent and concurrent. In non-concurrent environments, only one connection is active at any given time, and the node with the active connection owns the disk. Disk takeover occurs when the node that currently owns the disk fails and a surviving node assumes control of the shared disk so that the disk remains available.

In concurrent mode, the shared disks are actively connected to more than one node at the same time. Therefore, disk takeover is not required when a node fails. An additional product called the Concurrent Resource Manager is necessary to provide this functionality.

HACMP monitors and performs IP address takeover for the following TCP/IP-based communications adapters on cluster nodes: Ethernet, FDDI, and token-ring. IP address takeover occurs when a node assumes the IP address of a node that has failed. The takeover node can then provide the same network service that the failed node was providing to the cluster's clients. Additionally, it is possible for the MAC address of the adapter card to also fail-over to the standby adapter. This removes the necessity for the client population to flush and repopulate their ARP cache. The ARP cache contains the mapping of IP addresses to MAC addresses.

11.2.1.2 HAGEO overview

The IBM @server pSeries High Availability Geographic Cluster (HAGEO) system helps keep mission-critical systems and applications operational in the event of disasters, such as power outages or hardware and software failures. It does this by eliminating the system and the site as points of failure.

A HAGEO cluster consists of two geographically-separated sites, each capable of supporting up to four high-availability cluster system nodes. There are three modes of disaster protection and one mode of recovery: Remote hot backup, remote mutual takeover, concurrent access, and remote system recovery.

- **Remote hot backup** - A remote geographic site and system are designated as the hot backup site and system. The backup system includes hardware, system and application software, and application data and files. In the event of a failure, the failed system's application workload automatically transfers to the remote hot backup system.
- **Remote mutual takeover** - Geographically separated sites and systems can be designated as hot backups for each other. Mutual takeover allows each to operate as an independent system or as part of a distributed system. Should either experience a failure, the other acts as a hot backup and automatically takes over the designated application workload of the failed system.
- **Concurrent access** - Nodes at geographically-separated sites can have simultaneous access to the concurrent volume group and the same disk resources. If a node fails, the availability of the resource is not affected.
- **Remote system recovery** - After a failed information system site has been restored to operation, HAGEO can re-synchronize and reintegrate the failed system with the remote hot backup system. HAGEO updates the remote failed system with a current up-to-date mirror of application data and files processed by the backup system after the failed system ceased operations. Upon completing restoration of an up-to-date data and file mirror, the HAGEO cluster will resume synchronized system operations, including the mirroring of real-time data and files between the system sites.

HAGEO components

Figure 152 on page 512 shows a HAGEO scenario.

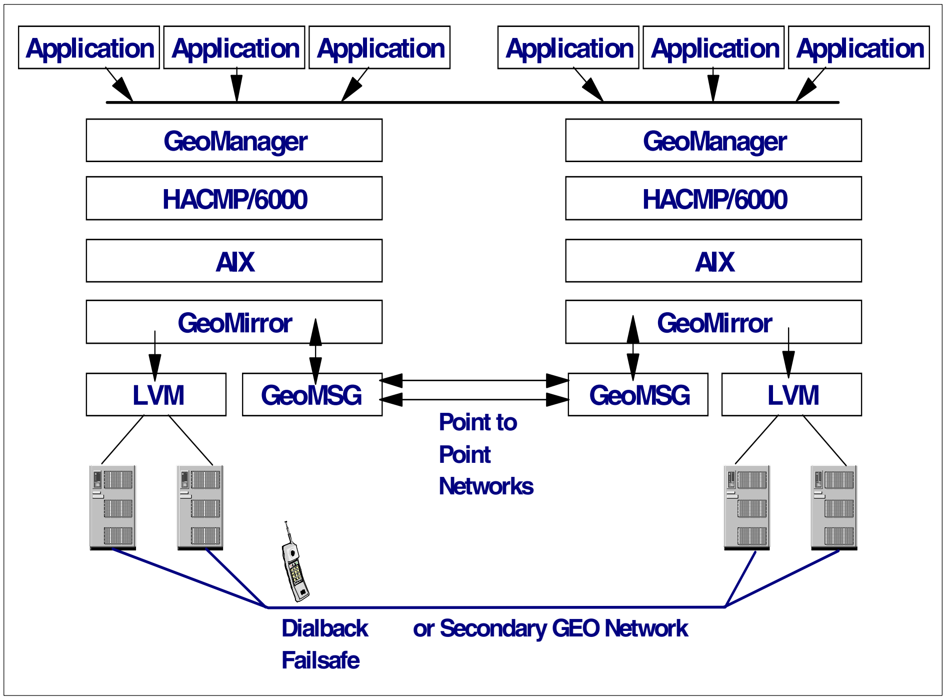


Figure 152. High availability geographic cluster

The GeoManager supervises the distributed cluster and adjusts the heart beat on long-distance communication links to detect site failure, and control failover; this helps avoid making the incorrect conclusion that the site has failed.

GeoMessage provides reliable communications and better performance by choosing the fastest path. It also provides load balancing across the links.

GeoMirror replicates data and coordinates updates removing the AIX limitation of three mirror copies of a disk and allowing three copies at each geographic site. It provides real-time data and file mirroring in three modes:

- **Synchronous mode** helps ensure that the same data exists at both sites at the completion of every write.
- **Synchronous with mirror write consistency** helps ensure that both sites can be restored with identical data, even in the event of a site failure in mid-transaction.

- **Asynchronous mode** writes on the local disk without waiting for the remote write to complete. This mode offers the best performance of the three on the local system node.

11.2.1.3 High-availability solutions

IBM now has packaged IBM @server pSeries High Availability packages with some of their enterprise server range. Each package has two highly-scalable IBM @server pSeries servers that are rack mounted in a 19 inch S00 rack with an external Serial Storage Architecture (SSA) subsystem with eight SSA disks. Also included is AIX Version 4.3, HACMP Version 4.3, and the equipment required for the servers to be connected in a high-availability solution. At the time of writing there are no announced high-availability solutions with AIX 5L.

The following are the hardware choices for the solution:

- HAB80 - each B80 (also known as the pSeries 640 system) is capable of 4-way 375 MHz SMP (Power 3-II) processing and 16 GB of memory
- HAH70 - each H70 is capable of 4-way 340 MHz SMP (PowerPC RS64 II) processing and 8 GB of memory.
- HAH80 - each H80 is capable of 6-way 450 MHz SMP (PowerPC RS64 III) processing and 16 GB of memory.
- HAM80 - each M80 is capable of 8-way 500 MHz SMP (PowerPC RS64 III) processing and 32 GB of memory.
- HAS80 - each S80 is capable of 24-way 450 MHz (PowerPC RS64 II) processing and 64 GB of memory.
- HAS85 - each S85 (also known as the pSeries 680 system) is capable of 24-way 600 MHz processing and 96 GB of memory.

These high-availability solutions from IBM have an estimated 99.999 percent availability, less than six minutes of annual unplanned system hardware and software downtime.

Optional applications and middle-ware with pretested high-availability scripts for:

- Collaborative
- Database
- Enterprise availability management
- Enterprise resource planning
- Transaction monitor

11.3 Windows 2000 high availability

Along with the improved stability and reliability of Windows 2000 compared to Windows NT version 4.0, there are additional features available in the two enterprise versions of Windows 2000 that the standard server package does not have.

11.3.1 Network Load Balancing (NLB)

NLB acts as a front-end cluster to a multi-tiered network. It distributes IP traffic across a cluster of up to 32 servers (running Windows 2000 Advanced Server), thus, providing a single system image to the client. By distributing the traffic, it allows the redirection of requests to other servers if one is down due to failure or maintenance.

Some of the benefits of NLB are:

- Automatic detection of failed or offline computer
- Automatic network load redistribution if cluster set changes
- Workload recovery and redistribution within 10 seconds
- Handles inadvertent sub-netting and rejoining of the cluster network
- Can be remotely started, stopped, or controlled
- No specialized hardware is required

11.3.2 Cluster Service

Just as the NLB interface acts as a front-end cluster, the Cluster Service acts as a back-end cluster. It is there to monitor the health and provide high availability for standard applications and services, such as databases and file and print servers. It can automatically recover data and applications from many of the common types of failures.

Through the MMC, there is a graphical management console that allows administrators to visually monitor the status of all the resources in the cluster and to move workloads around by pointing and clicking.

The clustering service that is a part of Windows 2000 Advanced Server supports a two node cluster where as the Datacenter Server supports up to four nodes.

Other features of the Cluster Service are:

- **Support for rolling upgrades** - This means shorter service outages and no need to rebuild the cluster configuration.

- **Network failure recovery** - It detects different states for network failures and uses the appropriate fail over policy to determine whether or not to fail over the resource group.
- **Use of Plug and Play technology** - This automatically detects the removal or addition of network adapters, TCP/IP network stacks, and shared physical disks.

Using both clustering technologies together, you can create a highly available and scalable e-commerce application by deploying NLB across your front-end Web servers and clustering on your back-end database servers. This provides a nearly linear scalability without server or application-based single points of failure.

Appendix A. Special notices

This publication is intended to help systems engineers, IT architects, consultants in understanding AIX and/or Windows 2000 from an operating system perpective. It provides a functional analysis of AIX and Windows 2000. The information in this publication is not intended as the specification of any programming interfaces that are provided by AIX 5L and/or Windows 2000. See the PUBLICATIONS section of the IBM Programming Announcement for AIX 5L for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX
APPN
CICS
DB2
e (logo)®
IBM ®
Netfinity
OS/2
OS/400
POWERparallel
Redbooks
RISC System/6000
RS/6000
S/390
Service Director
SP1
System/390
VisualAge
WebSphere

AnyNet
AS/400
CICS/6000
DB2 Universal Database
ESCON
IPDS
Open Blueprint
OS/390
PAL
RACF
Redbooks Logo
RMF
S/370
SecureWay
SP
SP2
TCS
VTAM
Xstation Manager

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere.,The Power To Manage., Anything. Anywhere.,TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Appendix B. Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

B.1 IBM Redbooks

For information on ordering these publications see “How to get IBM Redbooks” on page 527.

- *AIX 5L and Windows 2000: Solutions for Interoperability*, SG24-6225
- *Printing for Fun and Profit under AIX 5L*, SG24-6018
- *AIX 5L Differences Guide Version 5.0 Edition*, SG24-5765
- *AIX Version 4.3 Differences Guide*, SG24-2014-02
- *AIX 64-bit Performance in Focus*, SG24-5103
- *RS/6000 Graphics Handbook*, SG24-5130
- *Beyond DHCP - Work Your TCP/IP Internetwork with Dynamic IP*, SG24-5280-01
- *AIX 5L Workload Manager (WLM)*, SG24-5977

B.2 IBM Redbooks collections

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at ibm.com/redbooks for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	**Collection Kit Number**
IBM System/390 Redbooks Collection	SK2T-2177
IBM Networking Redbooks Collection	SK2T-6022
IBM Transaction Processing and Data Management Redbooks Collection	SK2T-8038
IBM Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
IBM AS/400 Redbooks Collection	SK2T-2849
IBM Netfinity Hardware and Software Redbooks Collection	SK2T-8046
IBM RS/6000 Redbooks Collection	SK2T-8043
IBM Application Development Redbooks Collection	SK2T-8037
IBM Enterprise Storage and Systems Management Solutions	SK3T-3694

B.3 Other resources

These publications are also relevant as further information sources:

- Chokhani, Dr. Santosh, “Trusted Products Evaluation,” July 1992, *Communications of the ACM*.
- Chowdhry, Pankaj, “Win 2000 RC2: Insidiously Important,” September 30, 1999, *PC Week*.
- Soloman, David A., *Inside Windows NT, Second Edition*, Microsoft Press, 1998, ISBN: 1-5723-1677-2
- *Active Directory Technical Summary*, Microsoft White Paper
- *AIX Performance Monitoring and Tuning Guide*, SC23-2365
- *International Support in Windows 2000*, Microsoft White Paper
- *Microsoft’s Active Directory Overview*, Microsoft White Paper
- *The RS/6000 64-bit Solution*, IBM White Paper
- *Secure Networking Using Windows 2000 Distributed Security Services*, Microsoft White Paper
- *Windows 2000 Reliability and Availability Improvements*, Microsoft White Paper

The following manuals are only shipped with AIX 5L in hardcopy format and in softcopy format as part of the Base Documentation CD:

- *AIX 5L Version 5.1 Installation Guide*
- *AIX 5L Version5.1 Network Installation Management Guide and Reference*
- *AIX 5L Version 5.1 Quick Beginnings*
- *AIX 5L Version5.1 Quick Installation and Startup Guide*

The following is also product documentation and can only be purchased with the software product:

- Windows 2000 Server documentation

B.4 Referenced Web sites

These Web sites are also relevant as further information sources:

- `http://www.microsoft.com/windows2000/library/resources/reskit/default.asp`
- `http://www.microsoft.com/Windows/server/Overview/features/interop.asp`

- http://www.microsoft.com/isapi/redir.dll?prd=msdn&pver=6.0&ar=library
- http://www.microsoft.com/directx/homeuser/downloads/default.asp
- http://msdn.microsoft.com/downloads/sdks/platform/tpipv6.asp
- http://msdn.microsoft.com/certification/appspec.asp
- http://msdn.microsoft.com/winlogo/
- http://windowsupdate.microsoft.com
- http://service.boulder.ibm.com/asd-bin/doc/en_us/winntcl2/f-feat.htm
- http://service.boulder.ibm.com/asd-bin/doc/en_us/win95cl/f-feat.htm
- http://www.ibm.com/servers/aix/products/aixos/bonus/details.html
- http://www.ibm.com/servers/aix/products/ibmsw/list/
- http://www.ibm.com/servers/monterey/
- http://www.ibm.com/java/jdk/index.html
- http://www.ibm.com/java/jdk/118/index.html
- http://www.ibm.com/java/jdk/rmi-iiop/index.html
- http://www.ibm.com/java/jdk/decimal
- http://www.ibm.com/java/jdk/aix
- http://www.software.ibm.com/websphere/appserv
- http://www.software.ibm.com/websphere/httpservers/
- http://www.software.ibm.com/enetwork/techexplorer
- http://www.software.ibm.com/enetwork/commserver/
- http://www.software.ibm.com/software/techexplorer
- http://www-4.ibm.com/software/network/directory/
- http://www.rs6000.ibm.com/resource/technology/64bit6.html
- http://www.rs6000.ibm.com/resource/technology/aixflyer.pdf
- http://www.austin.ibm.com/software/Standards/
- http://www.developer.ibm.com/java/j2/index.html
- http://www.projectmonterey.com/
- http://www3.innosoft.com/ldapworld/ldapfaq.html
- http://www.opengroup.org/
- http://www.cert.org/
- http://www.first.org/

- http://www.phrack.com/
- http://www.checkpoint.com/
- http://www.tivoli.com/
- http://www.fish.com/cops/
- http://www.webtrends.com/
- http://www.cybg.com/
- http://support.cai.com/arcservesupp.html
- http://www.legato.com/Products/index.html
- http://www.veritas.com/us/products/bent2000/index.shtml
- http://www.gated.merit.edu
- http://www.ietf.org/rfc.html
- http://www.ietf.org/html.charters/ipngwg-charter.html
- http://home.netscape.com/communicator/v4.5/index.html
- http://java.sun.com/products/java-media/jmf
- http://www.graphon.com/index.html
- http://www.adobe.com/acrobat/
- http://www.chilisoft.com/
- http://tarantella.sco.com/tsz
- http://www.apache.org/
- http://www.samba.org/
- http://www.netcs.com
- http://www.novell.com/press/archive/1997/07/pr97109.html
- http://www.internic.net
- ftp://ftp.porcupine.org/pub/security/tcp_wrappers_7.6.tar.gz
- ftp://ftp.porcupine.org/pub/security/satan-1.1.1.tar.Z
- http://www.haystack.com
- http://www.javasoft.com/products/javacomm
- http://www.rsasecurity.com/rsalabs/pkcs/pkcs-11
- http://www.ntp.org
- http://www.redbooks.ibm.com
- http://www.ebsinc.com/solaris/network/nis.html

- `http://www.ibmlink.ibm.com`
- `http://iamexwiwww.unibe.ch/system/CDE`
- `http://www.elink.ibmlink.ibm.com/pbl/pbl`

How to get IBM Redbooks

This section explains how both customers and IBM employees can find out about IBM Redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** ibm.com/redbooks

  Search for, view, download, or order hardcopy/CD-ROM Redbooks from the Redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

  Redpieces are Redbooks in progress; not all Redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

  Send orders by e-mail including information from the IBM Redbooks fax order form to:

	e-mail address
In United States or Canada	pubscan@us.ibm.com
Outside North America	Contact information is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	Country coordinator phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Fax Orders**

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	Fax phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and Redbooks by accessing the IBM Intranet Web site at http://w3.itso.ibm.com/ and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at http://w3.ibm.com/ for redbook, residency, and workshop announcements.

IBM Redbooks fax order form

Please send me the following:

Title	Order Number	Quantity

First name Last name

Company

Address

City Postal code Country

Telephone number Telefax number VAT number

□ Invoice to customer number

□ Credit card number

Credit card expiration date Card issued to Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Abbreviations and acronyms

ABI	Application Binary Interface
ACE	Access Control Entries
ACL	Access Control List
ADSM	ADSTAR Distributed Storage Manager
AFS	Andrew File System
AIX	Advanced Interactive eXecutive
ANSI	American National Standards Institute
APA	All Points Addressable
API	Application Programming Interface
APPC	Advanced Program-to-Program
APPN	Advanced Peer-to-Peer Networking
ARC	Advanced RISC Computer
ARPA	Advanced Research Projects Agency
ASCII	American National Standard Code for Information Interchange
ATE	Asynchronous Terminal Emulation
ATM	Asynchronous Transfer Mode
AVI	Audio Video Interleaved
BDC	Backup Domain Controller
BIND	Berkeley Internet Name Domain
BNU	Basic Network Utilities
BOS	Base Operating System
BRI	Basic Rate Interface
BSD	Berkeley Software Distribution
BSOD	Blue Screen of Death
BUMP	Bring-Up Microprocessor
CA	Certification Authorities
CAL	Client Access License
C-SPOC	Cluster single point of control
CDE	Common Desktop Environment
CDMF	Commercial Data Masking Facility
CDS	Cell Directory Service
CERT	Computer Emergency Response Team
CGI	Common Gateway Interface
CHAP	Challenge Handshake Authentication
CIDR	Classless InterDomain Routing
CIFS	Common Internet File System
CMA	Concert Multi-threaded Architecture
CO	Central Office
COPS	Computer Oracle and Password System
CPI-C	Common Programming Interface for Communications
CPU	Central Processing Unit
CSNW	Client Service for NetWare
CSR	Client/server Runtime

DAC	Discretionary Access Controls
DARPA	Defense Advanced Research Projects Agency
DASD	Direct Access Storage Device
DBM	Database Management
DCE	Distributed Computing Environment
DCOM	Distributed Component Object Model
DDE	Dynamic Data Exchange
DDNS	Dynamic Domain Name System
DEN	Directory Enabled Network
DES	Data Encryption Standard
DFS	Distributed File System
DHCP	Dynamic Host Configuration Protocol
DLC	Data Link Control
DLL	Dynamic Load Library
DS	Differentiated Service
DSA	Directory Service Agent
DSE	Directory Specific Entry
DNS	Domain Name System
DTS	Distributed Time Service
EFS	Encrypting File Systems
EGID	Effective Group Identifier
EISA	Extended Industry Standard Architecture
EMS	Event Management Services
EPROM	Erasable Programmable Read-Only
ERD	Emergency Repair Disk
ERP	Enterprise Resources Planning
ERRM	Event Response Resource Manager
ESCON	Enterprise System Connection
ESP	Encapsulating Security Payload
EUID	Effective User Identifier
FAT	File Allocation Table
FDDI	Fiber Distributed Data Interface
FDPR	Feedback Directed Program Restructure
FIFO	First In/First Out
FIRST	Forum of Incident Response and Security
FQDN	Fully Qualified Domain Name
FSF	File Storage Facility
FTP	File Transfer Protocol
FtDisk	Fault-Tolerant Disk
GC	Global Catelog
GDA	Global Directory Agent
GDI	Graphical Device Interface
GDS	Global Directory Service
GID	Group Identifier
GL	Graphics Library
GSNW	Gateway Service for NetWare
GUI	Graphical User Interface

HACMP	High Availability Cluster Multiprocessing
HAL	Hardware Abstraction Layer
HCL	Hardware Compatibility List
HSM	Hierarchical Storage Management
HTTP	Hypertext Transfer Protocol
IBM	International Business Machines Corporation
ICCM	Inter-Client Conventions Manual
IDE	Integrated Drive Electronics
IDL	Interface Definition Language
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IIS	Internet Information Server
IKE	Internet Key Exchange
IMAP	Internet Message Access Protocol
I/O	Input/Output
IP	Internet Protocol
IPC	Interprocess Communication
IPL	Initial Program Load
IPsec	Internet Protocol Security
IPX	Internetwork Packet eXchange
ISA	Industry Standard Architecture
ISDN	Integrated Services Digital Network
ISNO	Interface-specific Network Options
ISO	International Organization for Standardization
ISS	Interactive Session Support
ISV	Independent Software Vendor
ITSEC	Initial Technology Security Evaluation
ITSO	International Technical Support Organization
ITU	International Telecommunications Union
IXC	Inter Exchange Carrier
JFS	Journaled File System
JIT	Just-In-Time
L2F	Layer 2 Forwarding
L2TP	Layer 2 Tunneling Protocol
LAN	Local Area Network
LCN	Logical Cluster Number
LDAP	Lightweight Directory Access Protocol
LFS	Log File Service (Windows NT)
LFS	Logical File System **(**AIX**)**
LFT	Low Function Terminal
JNDI	Java Naming and Directory Interface
LOS	Layered Operating System

LP	Logical Partition
LPC	Local Procedure Call
LPD	Line Printer Daemon
LPP	Licensed Program Product
LRU	Least Recently Used
LSA	Local Security Authority
LTG	Local Transfer Group
LUID	Login User Identifier
LV	Logical Volume
LVCB	Logical Volume Control Block
LVDD	Logical Volume Device Driver
LVM	Logical Volume Manager
MBR	Master Boot Record
MCA	Micro Channel Architecture
MFT	Master File Table
MIPS	Million Instructions Per Second
MMC	Microsoft Management Console
MOCL	Managed Object Class Library
MPTN	Multi-protocol Transport Network
MS-DOS	Microsoft Disk Operating System
MSS	Maximum Segment Size
MWC	Mirror Write Consistency
NBC	Network Buffer Cache
NBF	NetBEUI Frame
NBPI	Number of Bytes per I-node
NCP	NetWare Core Protocol
NCS	Network Computing System
NCSC	National Computer Security Center
NDIS	Network Device Interface Specification
NDS	NetWare Directory Service
NETID	Network Identifier
NFS	Network File System
NIM	Network Installation Management
NIS	Network Information System
NIST	National Institute of Standards and Technology
NLS	National Language Support
NNS	Novell Network Services
NSAPI	Netscape Commerce Server's Application
NTFS	NT File System
NTLDR	NT Loader
NTLM	NT LAN Manager
NTP	Network Time Protocol
NTVDM	NT Virtual DOS Machine
NVRAM	Non-Volatile Random Access Memory
NetBEUI	NetBIOS Extended User Interface
NetDDE	Network Dynamic Data Exchange
OCS	On-Chip Sequencer
ODBC	Open Database Connectivity

ODM	Object Data Manager
OLTP	OnLine Transaction Processing
OMG	Object Management Group
ONC	Open Network Computing
OS	Operating System
OSF	Open Software Foundation
PAL	Platform Abstract Layer
PAM	Pluggable Authentication Module
PAP	Password Authentication Protocol
PBX	Private Branch Exchange
PCI	Peripheral Component Interconnect
PCMCIA	Personal Computer Memory Card
PDC	Primary Domain Controller
PDF	Portable Document Format
PDT	Performance Diagnostic Tool
PEX	PHIGS Extension to X
PFS	Physical File System
PHB	Per Hop Behavior
PHIGS	Programmer's Hierarchical Interactive Graphics System
PID	Process Identification Number
PIN	Personal Identification Number
PMTU	Path Maximum Transfer Unit
POP	Post Office Protocol
POSIX	Portable Operating System Interface for Computer Environment
POST	Power-On Self Test
PP	Physical Partition
PPP	Point-to-Point Protocol
PPTP	Point-to-Point Tunneling Protocol
PReP	PowerPC Reference Platform
PSN	Program Sector Number
PSSP	Parallel System Support Program
PV	Physical Volume
PVID	Physical Volume Identifier
QoS	Quality of Service
RACF	Resource Access Control Facility
RAID	Redundant Array of Independent Disks
RAS	Remote Access Service
RFC	Request for Comments
RGID	Real Group Identifier
RISC	Reduced Instruction Set Computer
RMC	Resource Monitoring and Control
RMSS	Reduced-Memory System Simulator
ROLTP	Relative OnLine Transaction Processing
ROS	Read-Only Storage
RPC	Remote Procedure Call

RRIP	Rock Ridge Internet Protocol
RSCT	Reliable Scalable Cluster Technology
RSM	Removable Storage Management
RSVP	Resource Reservation Protocol
SACK	Selective Acknowledgments
SAK	Secure Attention Key
SAM	Security Account Manager
SASL	Simple Authentication and Security Layer
SATAN	Security Analysis Tool for Auditing
SCSI	Small Computer System Interface
SDK	Software Developer's Kit
SFG	Shared Folders Gateway
SID	Security Identifier
SLIP	Serial Line Internet Protocol
SMB	Server Message Block
SMIT	System Management Interface Tool
SMP	Symmetric Multiprocessor
SMS	Systems Management Server
SNA	Systems Network Architecture
SNAPI	SNA Interactive Transaction Program
SNMP	Simple Network Management Protocol
SP	System Parallel
SPX	Sequenced Packet eXchange
SRM	Security Reference Monitor
SSA	Serial Storage Architecture
SSL	Secure Sockets Layer
SUSP	System Use Sharing Protocol
SVC	Serviceability
SWS	Silly Window Syndrome
TAPI	Telephone Application Program Interface
TCB	Trusted Computing Base
TCP/IP	Transmission Control Protocol/Internet Protocol
TCSEC	Trusted Computer System Evaluation
TDI	Transport Data Interface
TLS	Transport Layer Security
TOS	Type of Service
TTL	Time to Live
UCS	Universal Code Set
UDB	Universal Database
UDF	Universal Disk Format
UDP	User Datagram Protocol
UID	User Identifier
UMS	Ultimedia Services
UNC	Universal Naming Convention
UPS	Uninterruptable Power Supply

USB	Universal Serial Bus
UTC	Universal Time Coordinated
UUCP	UNIX to UNIX Communication Protocol
UUID	Universally Unique Identifier
VAX	Virtual Address eXtension
VCN	Virtual Cluster Name
VFS	Virtual File System
VG	Volume Group
VGDA	Volume Group Descriptor Area
VGSA	Volume Group Status Area
VGID	Volume Group Identifier
VIPA	Virtual IP Address
VMM	Virtual Memory Manager
VP	Virtual Processor
VPD	Vital Product Data
VPN	Virtual Private Network
VRMF	Version, Release, Modification, Fix
VSM	Virtual System Management
W3C	World Wide Web Consortium
WAN	Wide Area Network
WFW	Windows for Workgroups
WINS	Windows Internet Name Service
WLM	Workload Manager
WOW	Windows-16 on Win32
WWW	World Wide Web
WYSIWYG	What You See Is What You Get
WinMSD	Windows Microsoft Diagnostics
XCMF	X/Open Common Management Framework
XDM	X Display Manager
XDMCP	X Display Manager Control Protocol
XDR	eXternal Data Representation
XNS	XEROX Network Systems
XPG4	X/Open Portability Guide

Index

IBM Redbooks review

Your feedback is valued by the Redbook authors. In particular we are interested in situations where a Redbook "made the difference" in a task or problem you encountered. Using one of the following methods, **please review the Redbook, addressing value, subject matter, structure, depth and quality as appropriate.**

- Use the online **Contact us** review redbook form found at **ibm.com**/redbooks
- Fax this form to: USA International Access Code + 1 845 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Document Number **Redbook Title**	SG24-4784-02 AIX 5L and Windows 2000: Side by Side
Review	
What other subjects would you like to see IBM Redbooks address?	
Please rate your overall satisfaction:	O Very Good O Good O Average O Poor
Please identify yourself as belonging to one of the following groups:	O Customer O Business Partner O Solution Developer O IBM, Lotus or Tivoli Employee O None of the above
Your email address: The data you provide here may be used to provide you with information from IBM or our business partners about our products, services or activities.	 O Please do not use the information collected here for future marketing or promotional contacts or other communications beyond the scope of this transaction.
Questions about IBM's privacy policy?	The following link explains how we protect your personal information. **ibm.com**/privacy/yourprivacy/

IBM

Redbooks

AIX 5L and Windows 2000: Side by Side

AIX 5L and Windows 2000: Side by Side

Discover new features in AIX 5L and Windows 2000

Exploring fundamental cutting-edge technologies

Learn differencies and similarities between AIX 5L and Windows 2000

The object of this redbook is to demonstrate the AIX 5L and Windows 2000 platforms to show the reader similarities and differences between each operating system. Whether you are a Windows expert looking to learn more about the latest version of AIX, AIX 5L, or are an AIX expert and are looking to inform yourself of the latest Windows platform, Windows 2000, you will find each chapter in this redbook covers the fundamental technologies that make each operating system what it is.

In ensuing chapters, we will discuss fundamental operating system concepts, architectures, open standards compliances, and product packaging for both AIX 5L and Windows 2000. Then, we shall go into the user interfaces for both, storage management, security standards compliance and operations, and full systems management. Finally, we will give an in depth discussion of networking concepts on both platforms and demonstrate the full extent of scalability and high availability on both AIX 5L and Windows 2000.

Furthermore, while not much has changed in Windows 2000, AIX is relatively new at the time of writing this redbook, and we shall make a special point of pointing out the differences between AIX 5L and the previous version, AIX version 4.3.3.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-4784-02 ISBN 0738421626