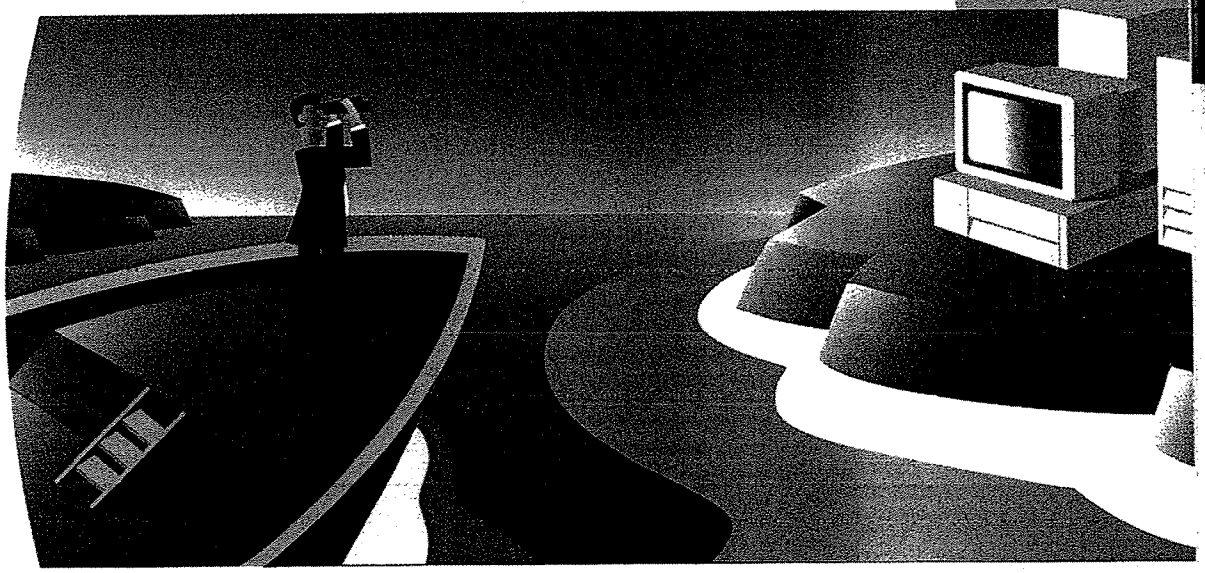
EXHIBIT D

Exploring IBM @server pSeries

"...explores both the machine's features and its place in the networks."

—*Data Communications*

TWELFTH EDITION



The Instant Insider's Guide to IBM's Family of UNIX Servers

Foreword by Adalio Sanchez, General Manager, @server pSeries, IBM Corporation

Jim Hoskins and Robert Bluethman

APLNDC-WH-A 0000019700

Publisher: Jim Hoskins

Manager of Finance/Administration: Joyce Reedy

Production Manager: Gina Cooke

Cover Designer: Lauren Smith

Copyeditor: Andrew Potter

Proofreader: Jacquie Wallace

Indexer: Susan Olason

Printer: P.A. Hutchison

Recognizing the importance of preserving what has been written, it is a policy of Maximum Press to have books of enduring value published in the United States printed on acid-free paper, and we exert our best efforts to that end.

Library of Congress Cataloging-in-Publication Data

Hoskins, Jim.

Exploring IBM eServer pSeries / Jim Hoskins, Robert Bluethman.— 12th ed.

p. cm.

ISBN 1-931644-04-7

1. IBM computers. 2. UNIX (Computer file) I. Bluethman, Robert, 1941- II. Title.

QA76.8.I1015H665 2003

004.2'525—dc21

2003014398

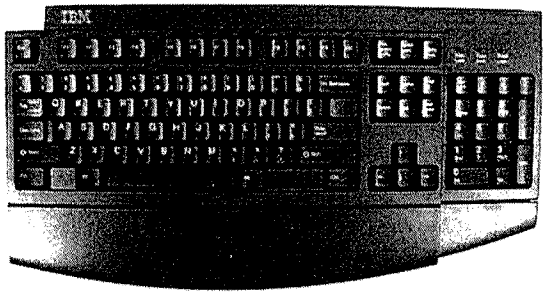


Figure 2.17. The IBM Quiet Touch Keyboard used with pSeries systems.

USB keyboards for all language groups are also available for pSeries systems, primarily for use with servers that support logical partitioning (LPAR). When it is necessary to configure more than one LPAR of a server with keyboard, mouse, and display, USB keyboards and the three-button mouse (#8841) are required. Since pSeries systems do not provide integrated USB ports (such as are common on personal computers today), a USB Keyboard/Mouse Attachment Card (#2737) must be installed in a PCI slot assigned to the appropriate LPAR. This PCI adapter provides the controller and USB port for one keyboard and a mouse, which plugs into the keyboard. A 2-meter USB keyboard attachment cable is used to connect the keyboard to the attachment card (#2737).

Modems

A modem ("modem" stands for "modulator-demodulator") is a device for modulating or translating digital computer signals into analog telephone line signals for transmission, and vice versa (demodulating) for receiving signals from other systems. IBM does not sell modems, but many industry-standard modems are available. With dial-up telephone lines, 56 KBps modems are commonly used. The modems most often used with pSeries systems are high-speed devices for use on leased telephone circuits.

3

Application Programs and Operating Systems

Types of Software—A Conceptual Model

The term "software" is analogous to the term "publication." Newspapers are a category of publication. Annual reports, novels, and Who's Who directories are some other categories of publications. These different categories fill very different needs. The same situation exists with software. There are three basic categories, or software layers—the application program layer, the operating system layer, and the device driver layer, as shown in Figure 3.1. Each software layer performs a completely different job, but all three work closely together to perform useful work for the user.

Application Programs

The top software layer in the model is the application program layer (highlighted in Figure 3.2). The programs in this layer "apply" pSeries systems to a specific task (computer-aided design, word processing, ac-

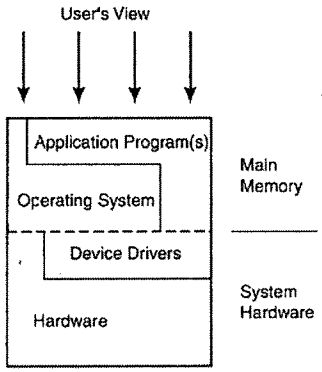


Figure 3.1. Conceptual software model of the pSeries system's basic software structure. The three layers of the software model work together to perform useful work for the user.

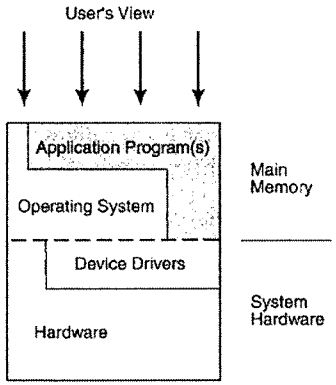


Figure 3.2. The application program software layer of the software model. It is the application program that defines the particular task the computer is performing for the user.

counting, etc.) and thus are called "application" programs. They actually perform the task for which the user purchased the computer while the other two layers play important support roles. A single pSeries system might run one application program at a time, or it might run many application programs simultaneously.

The arrows in the figure indicate how users conceptually "see" the computer system. The user usually interacts with the application program layer and (less frequently) the operating system layer. By working closely with the other software layers, the application program processes the various keystrokes made by the user and responds by displaying information on the computer's display or some other output device.

As we will see later in the chapter, many programs written for other computers that run UNIX operating systems (i.e., open systems) can be migrated to the pSeries and the AIX 5L operating system by the software developer. This allows pSeries users to capitalize on many application programs originally developed for other open systems. There is an application program to help users with just about anything they wish to do. Some examples of common functions that application programs provide include e-business (Web serving, e-mail, secure on-line shopping, customer support, etc.), accounting, statistical analysis, database, banking, telecommunications, and computer-aided design.

The application software layer is most familiar to the user because it provides a business function and is the layer with which he or she interfaces. However, the operating system and device driver layers are equally vital, since they provide the framework or "infrastructure" that controls the hardware (pSeries system and attached I/O devices) and allows multiple independent applications to execute. The interactions between the application and operating system layers, as well as those between the operating system and device driver layers, are very complex. Although it is important for programmers and systems administrators to understand these layers in detail, it is sufficient for most readers to think of the operating system and device drivers as a single entity that supports application software on one side and system hardware on the other. The next two sections give a high-level description of the specific roles of the operating system and device driver layers.

Operating Systems

The operating system (highlighted in Figure 3.3) must manage the hardware resources of the computer system and perform tasks under the con-

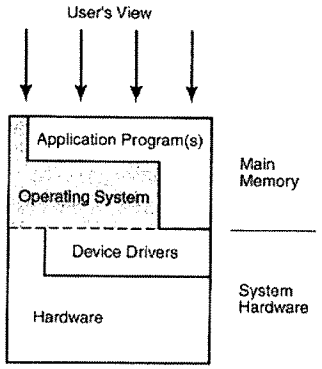


Figure 3.3. The operating system software layer of the software model. The operating system provides the environment in which the application programs run.

trol of application program(s) and keyboard/mouse commands issued by the user. The application program can rely on the operating system to perform many of the detailed housekeeping tasks associated with the internal workings of the computer. Thus, the operating system is said to provide the environment in which application programs execute. The operating system layer also accepts commands directly from the user; for instance, it can copy files, change a user's password, and so on.

The base operating system used by pSeries systems is AIX 5L. AIX 5L provides an environment that can run one application program at a time or many application programs simultaneously. There are also many extensions to the AIX 5L operating system, such as the graphics program OpenGL, that allow the user to customize the operating system environment. Because of the modular nature of AIX 5L these extensions plug right into and essentially become a part of the operating system. Some customers and third-party software vendors may write kernel extensions for use with their applications, but those extensions are not tested and supported by IBM, whereas the extensions provided by IBM are supported along with AIX 5L.

Although not distributed by IBM, the Linux operating system is supported on pSeries servers. At the time of this writing, Linux distributions from SuSE are supported on most pSeries servers, including LPARs. Distributions from Red Hat and Turbo Linux are available for selected pSeries and RS/6000 systems. This chapter includes a section on Linux support for pSeries servers. For links to additional information about Linux, see the companion Web site for this book *(www.maxpress.com)*.

Device Drivers

The third and final layer of software in our software model is the device driver layer (highlighted in Figure 3.4). "Device driver" is a fancy term for a set of highly specialized programs, usually written by the manufacturer of computer hardware. These specialized programs reside in pSeries main memory or in memory provided right on the adapters they control.

Unlike application programs or operating systems, device drivers are only used by other programs. That is, device drivers never interact directly with the users and exist only to help application programs and the operating system perform their tasks. They interact directly with computer hardware elements under the control of the operating system

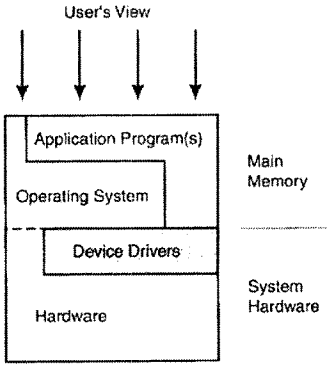


Figure 3.4. The device driver software layer of the software model. Device drivers directly control the hardware elements of pSeries systems and shield application programs and operating systems from hardware details.

or application program layers. Device drivers also help shield application programs from the hardware specifics of computers, allowing for evolutionary product improvements without sacrificing application program compatibility. Each hardware feature (device or adapter card) requires a device driver program. The device drivers for pSeries features (discussed in Chapter 2) are included with AIX 5L.

pSeries Software Compatibility

Computer systems that run the UNIX operating system have been in existence for many years. As a result, a wide variety of application programs have been developed for UNIX. The flexibility afforded by virtue of this large and diverse software base allowed computers running the UNIX operating system to fill many different needs. Of course, this sea of available application programs did not exist when the original UNIX operating system was first announced. It took the independent efforts of a great many people over many years to develop the large number of application programs (called an application program software base) that exist today.

Why Compatibility Is Important

In order to capitalize on that software base, application software compatibility was a primary objective in the design of the AIX 5L operating system. That is, most application programs written for UNIX operating systems can easily be migrated to the pSeries system and the AIX 5L operating system by the software developer.

It is important to understand that of the three software layers in our software model, compatibility with programs in the application programs layer is most important. Why? First of all, application programs typically represent the lion's share of a user's software investment. Further, being forced to abandon an application program because of incompatibilities may also make users throw away whatever data and training or experience they have accumulated with the application program—both of which can be substantial. Some users have developed custom application programs at considerable cost in development time and money. Incompatibility at the application program level would render these programs virtually useless. Last, and perhaps most important, application layer compatibility allows pSeries system users to choose from the thousands of application programs that were originally developed for the UNIX operating system.

What about the operating system and device driver layers? The ability to run earlier UNIX operating system software is not important for several reasons. Operating systems typically represent only a small fraction of the user's software investment; they can be upgraded easily without changing the user's view of the computer system. Further, a new operating system is usually necessary to allow users to have access to new features of the computer system not considered by the programmers of the old operating system. Of course, one of the primary purposes of the device driver layer is to allow the computer hardware to change without affecting compatibility with the operating system and application programs. This is done by changing the way a device driver interacts with the hardware without changing the way the device driver interacts with the operating system or application programs. The user is supplied with new device drivers to support new pSeries hardware.

Inside pSeries Compatibility

To understand pSeries compatibility, you must understand a little about how application programs are written. First, a programmer writes an application program's instructions or code using a programming language (e.g., C++, FORTRAN, etc.). A programming language is basically a library of computer instructions (easily understood by programmers) from which a programmer may choose to write programs. The list of programming language instructions that make up an application program is called the program's source code. In order to run the application program on a computer, the source-code instructions must be converted into instructions a computer can understand. This conversion process is called compiling the program. When the source code is compiled, the result is called object code and can be directly executed by the computer hardware.

The most important thing to understand about pSeries compatibility is that any model of the pSeries family is object-code-compatible with all other pSeries models and earlier RS/6000 systems. That is, you can take the object code for an application program compiled on an

RS/6000 Model 150, load it on a properly configured pSeries 630, and run the program with no changes.

A word of caution is necessary at this point. Object-code compatibility as just described is true for 32-bit applications (i.e., applications written for systems that used 32-bit microprocessors). All current pSeries and RS/6000 computers, except RS/6000 Models B50 and 150, use 64-bit architectures, and they can run AIX 5L applications written for earlier 32-bit systems. However, the reverse is not true. Applications written to take advantage of 64-bit architectures cannot be run on older systems that do not have 64-bit capability. This situation is called "upward compatibility," since it enables users to carry forward their investment in applications on earlier systems to run on the latest, more powerful computers.

Since AIX 5L is compliant with UNIX 98 standards, most other application programs written for UNIX operating systems, including earlier versions of AIX, and for Linux, will be highly source-code-compatible with pSeries systems. This means that a programmer must make only a few minor changes in the application program source code, load the source code written for the UNIX operating system on a pSeries system, and then recompile the source code using the appropriate pSeries compilers. The new object code generated can then be executed on the pSeries system. This process is called "porting."

In addition to UNIX source-code compatibility, AIX 5L provides "interoperability" functions that facilitate systems management in mixed environments of pSeries systems and UNIX servers from other suppliers. These functions include UNIX System V.4 commands, hence the term "SVR4 affinity." SVR4 affinity tools make AIX 5L a bit more familiar to information technology (IT) personnel who have experience with UNIX operating systems other than AIX 5L.

For applications written for Linux, AIX 5L provides interfaces and software libraries to make it as easy as possible to port those applications to run on pSeries systems. These "Linux affinity" tools will be described later in the chapter. As described in Chapter 1, native Linux distributions are available for most pSeries systems, and Linux can run in one or more logical partitions on p630, p650, p655, p670, and p690 servers.

The syntax or grammar of source code for various languages is defined by independent standards bodies. This compliance to industry standards is what makes an "open system" open and is the essential element in pSeries software compatibility.

Application Programs

The previous discussion covered the three basic software layers in pSeries systems that cooperate to perform useful work for the user. Now we turn our attention to the top layer of our software model—application programs (see Figure 3.2, shown previously). It is the application program that actually "applies" the pSeries system's computational power to a particular task. Some companies choose from available application programs designed, written, and sold by various software suppliers. These are called prewritten application programs. Other businesses design and write their own custom application programs or use a combination of prewritten and custom application programs. Although some basic types of application programs will be discussed, this chapter is by no means a comprehensive guide to application programs.

Application programs range from simple programs that concentrate on a very specific task to powerful and very complex groups of programs—designed to work together. They perform a myriad of functions as diverse as the environments in which you find computers today. Every person who uses a personal computer (PC) is familiar with prewritten application programs—word processors, spreadsheets, databases, graphics presentation packages, and Web browsers. These are known as cross-industry application programs because they provide basic tools that are used by people in every type of enterprise to communicate with one another and to manage information. pSeries systems are targeted to the server marketplace and are seldom used as individual workstations. Users of pSeries systems typically run these common personal productivity applications on a PC attached to the server on a local area network (LAN).

Just as word-processor and graphics presentation programs are important tools for individuals to use on a personal computer for documenting ideas and information and for communicating with others, there are many applications available for use on pSeries servers. Some applications, such as databases, are useful in almost every type of business. Database applications store information in a structured way and allow access in a variety of formats or "views," depending on the needs of the user. For readers who are not familiar with databases, the following simple illustration will help explain how they work.

To deal with large amounts of information efficiently, it is necessary to organize the information in a uniform manner. For example, the information in a telephone book is organized into an alphabetical list of names, addresses, and telephone numbers.

Database programs organize information into files, records, and fields. Don't be intimidated by the words. This is exactly how the information in a phone book is structured. Figure 3.5 shows an example telephone book listing and the corresponding computer database structure. The phone book itself is analogous to a file or set of information, also called a database. The information about one person in the phone book would be analogous to a record. The records contain the information for a given entry, and each record contains the same information about its respective entry. In this case, a record would contain the name, address, and phone number of the person. Each of these three items would be analogous to a field within a record. For example, the address part of a phone book entry would be called the address field.

Manually looking up information in a phone book quickly becomes fatiguing. The same is true for manually manipulating any large body of information. Once the information is entered into a database application program, however, it can be retrieved quickly and easily. Databases can contain information about a store's inventory, a library's books, personnel records, medical records, or virtually any other type of information. Organizations such as banks, airlines, and insurance companies commonly use extremely large databases shared by many users. Database application programs provide programming languages with which users can customize their database environments and develop the interfaces or views needed by various departments within a company or by their customers.

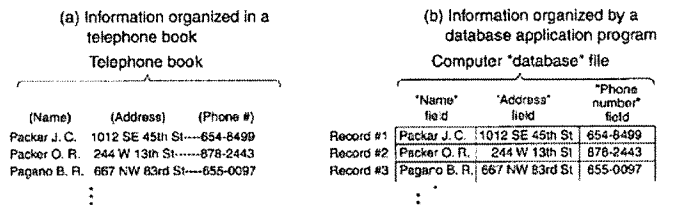


Figure 3.5. (a) The information structure used in a telephone book. (b) The same information organized into a database structure. To efficiently manipulate large amounts of information, it is necessary to first organize the information into a consistent format. The organization used by database application programs is not unlike that used in a telephone book.

pSeries systems provide the disk storage capacity, the communications interfaces, and the processing performance required to implement large databases and to support many users making queries of those databases. Products such as IBM DB2® Universal Database (UDB) and Oracle 9i Database are often used in conjunction with other applications to provide complete industry-specific solutions. Since the database software is used between the end-user application program and the AIX 5L operating system, databases are sometimes called "middleware" products. Middleware products are those which work with the operating system to compose the infrastructure needed for a particular application environment.

Industry-Specific Application Programs

Most companies will need specialized software designed for their particular industry or environment. These "industry-specific" application programs are written to address the needs of a well-defined environment. A research lab office has different application program needs than a dental practice. Such prewritten industry-specific application programs allow pSeries systems to be used effectively in highly specialized business, professional, and scientific environments.

There are applications designed for market sectors like banking, manufacturing and distribution, insurance telecommunications, Internet Service Providers (ISPs), retail, government and the public sector, healthcare, scientific research, petrochemicals, and research organizations—to name only a few. Many software vendors offer modular software products, based on a single underlying software design, that can be tailored to the needs of various industries or types of users. This approach allows, for example, both a manufacturer and a healthcare provider to manage products or services and customer relationships using essentially the same software product—but with user interfaces and specific functions suitable for their unique business environment. Users select only the software modules needed, which helps them control costs. Since the underlying software design is used across all the industry-specific solutions, all customers benefit from improved software quality and stability.

It would be impractical to list in this book all the business areas in which pSeries servers are used—and almost impossible to identify all the software suppliers who offer products for each category. Readers who are interested in that level of detail can refer to the companion Web site for this book *(www.maxpress.com)* for links to applications

software for pSeries systems. Some of the key applications areas in which pSeries servers are used include enterprise resource planning (ERP), supply chain management (SCM), customer relationship management (CRM), banking, retail, science and technology, e-business/e-commerce, and business intelligence (BI).

There is a lot of overlap among these categories, since nearly every business or enterprise has some of the same computing requirements. For example, almost all organizations have customers or clients to support, and many have supplier interactions to manage. Today, almost every organization must manage Web-based interfaces for customers and employees. Finally, many different kinds of customers need real-time data to measure levels of service, orders and deliveries, and inventory levels.

Custom Application Programs

Although cross-industry and industry-specific application programs fit many needs, other users require software designed to perform unusual and specific tasks or to conform with existing procedures. In these cases, it is often better to develop custom application programs written to exact specifications.

Custom application programs may be written by programmers within the company or by an outside consulting firm. In either case, the basic development steps are first to write a software specification that describes what the program will do. Then a preliminary version of the program is written that demonstrates the function that eventually will be in the final program. This preliminary version is evaluated by the user, and the specification is altered to reflect any needed changes. Then the final program is written and installed at the user's location. Typically, training will be provided by the developer and any problems ironed out. Once the user accepts the program, the software then has to be supported. That is, users will need a place to go when they have questions not addressed by the manuals. Support also includes making necessary changes to the application program as the user's needs change over time. This kind of ongoing support is critical to the success of any computer automation project.

Custom application program development is expensive and time-consuming, compared to the prewritten application program approach. But some companies may find that this additional expense and time can be recovered by the increased productivity and competitive advantage

that result from custom applications that precisely fit their needs. For customers who choose the custom software approach, pSeries systems and the programming tools available for the AIX 5L operating system provide a very productive software development environment.

Operating System

Few areas in information processing create more confusion and apprehension than the operating system layer of our software model, shown in Figure 3.3. The following discussion will help remove some of the mystery associated with the Advanced Interactive eXecutive (AIX) operating system used with all pSeries and RS/6000 systems. The reader will become familiar with basic operating system terms such as "interactive processing" and "multiuser," and then move in for a closer look at the AIX 5L operating system.

The description of AIX in this chapter applies to AIX 5L for POWER Version 5 Release 1 (AIX 5L v5.1) and Release 2 (AIX 5L v5.2). AIX Version 4 Release 3 (AIX 4.3) has been withdrawn from marketing and service will be discontinued December 31, 2003. Users of existing pSeries servers can upgrade to AIX 5L, and only AIX 5L is available with new systems. All earlier AIX versions and releases have been withdrawn as newer versions have been announced. In general, IBM supports two levels of AIX at any time, the latest level (version N) and the prior level (version N-1), which at this time are AIX 5L v5.2 and v5.1.

Introduction to Operating System Concepts

The operating system provides the necessary interface that allows the user and application programs to interact with pSeries systems. The user can interact directly with the operating system's user interface to manage files on a disk, start application programs, print files, and so on. The operating system also performs tasks directly under the control of application programs without any user assistance. The application program initiates tasks by directly interacting with the operating system through the application program interface (API). This is simply a set of operating system commands that can be issued directly by the application program. The API simplifies the job of the application programmer, who need not get involved with the details of hardware interaction. Further, when an application program uses the API, it is shielded from changes in

the computer hardware as new computers are developed. That is, the operating system (and device drivers) can be changed to support new computer hardware while preserving the API unchanged and allowing application programs to run unchanged on the new computer.

To understand the job of the operating system, it is necessary to understand a few basic concepts:

- Batch versus interactive processing
- Multiuser
- Multitasking
- Multiprocessing

Batch Versus Interactive Processing

There are two basic types of work a computer can perform: batch processing and interactive processing. To understand the difference between these two concepts, let's use an analogy and examine the difference between communications through the postal service and through the telephone.

If you wish to ask a distant friend some questions, you can either write a letter or phone. With the first option you gather all of your thoughts, put them on paper, and then submit the letter to a mailbox. A few days later (assuming your friend is responsive) you go to your mailbox and get the responses to your questions in the form of a document. This is analogous to batch processing with a computer in that you submit a request for the computer to answer some question(s) or perform some task(s). Some time later (from minutes to days), you can go to the printer and get the computer's responses in the form of a report. In the early days of computing, batch processing was the only alternative for computer interaction. Today, batch processing still has its place, but interactive processing is the norm.

Moving back to our analogy, sometimes you can't simply write down your list of questions in a letter because some of the questions you have will depend on the answer to one or more initial questions. In this case, you either have to send several letters back and forth between yourself and your friend or call your friend on the phone. Having a conversation with your friend over the phone is analogous to interactive processing

on a computer. With interactive processing, you have a dialogue with the computer system from a PC. You enter questions or requests for activity, and the computer immediately responds. So interactive processing gives the user an immediate response, which is required and expected in most applications (e.g., airline reservations or a retail checkout line).

Some applications use a combination of batch and interactive processing. For example, a payroll clerk might enter information from time cards in a dialogue style (interactive processing). Once all time cards are entered and verified to be correct, the clerk then issues a command that tells the computer to print all checks (a batch job). The clerk would later get the checks from the printer. AIX, like other operating systems, supports both batch and interactive processing.

What Is Multiuser?

A computer system is said to be a multiuser system if two or more users can share the system at the same time. With a multiuser computer system, from two to many hundreds of computer terminals are attached to a single computer. Each terminal provides its user with a "window" into the computer system and allows the user to perform tasks independent of all other users. Although the single computer system is being used simultaneously by many users, each user is unaware of the activities of the other users and seems to have his or her own computer system. However, a user may see the computer slow down (increase response time) as more and more users sign on to the computer and start doing work.

A multiuser computer system has several advantages over single-user systems. Because the computer system hardware and programs can be simultaneously shared by many users, no one has to stand in line waiting for a turn on the computer. Everyone (assuming there are enough terminals attached) has access to the computer whenever it is needed to do a job. Other advantages offered by a multiuser system are in the areas of security, accounting, backup/recovery, and so on. The pSeries running the AIX 5L operating system can act as either a single-user or a multiuser computer system.

What Is Multitasking?

Many people confuse multitasking with the term "multiuser" just discussed. As was just noted, "multiuser" refers to the capability to share a

single computer system between two or more users simultaneously. "Multitasking," also called "multiapplication," refers to the ability to simultaneously run two or more independent application programs for a single user. The opposite of multitasking is single tasking, which means that the computer user must finish using one application program before another can be started. The AIX operating system used with pSeries systems supports a full multitasking environment.

Multitasking is helpful in many environments. The office, for example, is one environment in which workers are often interrupted in the middle of one task to perform another. The multitasking capability of operating systems fits naturally into this interrupt-driven environment by allowing users to easily switch back and forth between several simultaneously active application programs as the interruptions occur. Another advantage afforded by multitasking is the ability to have the computer system perform batch-processing tasks while one is working with some other application program. This type of batch processing is called background processing. With background processing, a programmer can start a compile (batch processing) and then immediately go to work on some other program. An engineer can start a finite element analysis of a mechanical design (batch processing) and then immediately begin working on another design project. Without the background-processing capability provided through multitasking, the user would have to wait for the batch processing to complete before going on to any other tasks.

What Is Multiprocessing?

Whereas "multiuser" refers to the ability for many users to share the resources of a system concurrently and "multitasking" means that the system can run many different applications simultaneously, the term "multiprocessing" refers to the ability of an operating system to support many processors, all of which share system resources such as memory and input/output devices. A multiprocessor-capable operating system, such as AIX, allocates work to the available processor(s), thus increasing the performance of the system above what would be possible if there were only one processor available. The user does not have to worry about which specific processor is running his or her application. In this book we have described many pSeries models as symmetric multiprocessing (SMP) systems, and you should have noticed that the larger the system (i.e., the more processors), the higher the performance.

The word "symmetric" simply means that all the processors share memory and I/O devices equally.

In an SMP system, it is possible for one application to "hog" all the system resources. To overcome that exposure, AIX Workload Manager is an important software tool that allows the systems administrator to allocate a certain amount of processing power for specific applications, thus ensuring a base level of processing power for each major application. So, if the system is very busy, an application will always get at least a predetermined level of processing power. And, if the system is less heavily loaded, that application may run even faster, since the operating system can allocate more processing power to it.

What Is Logical Partitioning?

As explained in the section on logical partitioning in "A Closer Look" in Chapter 1, logical partitioning is really a capability of the pSeries hardware, namely the POWER4 and POWER4+ processors and the system firmware or microcode. LPAR allows the pSeries system to be partitioned into multiple "virtual servers within a server," each having at least one or maybe several processors assigned to it and each LPAR running its own copy of the operating system. The allocation of processors, memory, and I/O is flexible and does not have to conform to physical boundaries, such as how many processors are on a multichip module (MCM) or which PCI bus the I/O slots are on. The number of LPARs depends on the size of the pSeries server. A p690 can have 32 LPARs, while a p670 can have up to 16, a p650 up to eight, and the p655 and p630 up to four LPARs.

Although not an operating system function, LPAR is supported and enhanced by the operating system in several ways. On pSeries servers, each partition can run any of three different operating systems—AIX 5L v5.1, AIX 5L v5.2, or Linux. The functions available in each "virtual server" depend on the operating system. Each LPAR running AIX 5L v5.1 or v5.2 will have the support of all the autonomic computing and systems management capabilities of AIX 5L. But since Linux does not currently support all the functions available in AIX 5L, LPARs running Linux will be limited to the functions provided by that operating system. Two examples of such functional differences are dynamic deallocation of processors and PCI buses (which help increase system availability) and AIX Workload Manager (which allows the administrator to assign resources to various application workloads). For LPARs

running AIX 5L, these capabilities work just as they do on a pSeries SMP server without LPAR, but LPARs running Linux will lack those functions. On the other hand, the reason for running Linux in some LPARs is to have compatibility with the Linux applications the user requires and, presumably, those applications will run just fine without the AIX 5L capabilities.

AIX 5L v5.2 enhances LPAR with the ability to dynamically add or remove processors, memory, or adapters to a partition without rebooting the LPAR or the system. In order to reassign resources to LPARs running AIX 5L v5.1 or Linux, the partition must be restarted (static LPAR). Also, the dynamic LPAR capability provided by AIX 5L v5.2 works with processor and memory CUoD (Capacity Upgrade on Demand) to allow newly activated processors or memory to be assigned immediately to the appropriate LPARs and to allow dynamic processor sparing.

Logical partitioning offers the user many advantages. For example, an application or set of applications can be run in a specific logical partition, guaranteeing the availability of the assigned resources for that workload. Since different applications can run in each LPAR, one possible scenario is to have a production LPAR running the tested software version of an application, while in another LPAR there is a test system running a new version of the application. Another example is to run regional workloads in different LPARs, each with its correct date and time setting for the users in the region being supported. All of these usage scenarios provide cost-effectiveness advantages in systems management and physical space, since a single system is being used where previously the only alternative solution was to run multiple smaller systems.

AIX 5L—An Executive Overview

The UNIX operating system was originally developed by AT&T's Bell Labs in 1969. Over the years, it has continually been enhanced by various independent organizations (both academic and business), often in a nonstructured way. These independent efforts have resulted in a very powerful, somewhat cryptic, often awkward, and usually flexible operating system. The UNIX operating system became popular for several reasons. First, AT&T licensed the operating system to many different computer manufacturers, which offered the UNIX operating system for their computers rather than choosing to write their own operating systems. Second, the UNIX operating system is prevalent in colleges and universities. As students learned the UNIX operating system in these academic environments, it was only natural that they would seek out such systems after entering the nonacademic world. Finally, the C programming language fostered by the UNIX operating system is a highly popular one, offering very flexible and powerful programming structures.

The UNIX operating system has evolved to be the basis for the open-system marketplace, in which compatibility with industry standards is the hallmark. Compatibility means that programs written for one brand of "open system" can easily be migrated to another brand. The open-system concept is good for application program developers because they can easily offer their products on many different brands of computers. This is good for users because they have a multitude of software from which to choose no matter which brand of open system they buy.

IBM took the basic UNIX operating system and incorporated many enhancements developed by other organizations (e.g., the University of California at Berkeley), added many enhancements of its own, and came out with its version, called the Advanced Interactive eXecutive. This open-system approach was a departure from IBM's proprietary systems such as the OS/400 operating system for AS/400 systems or the MVS operating system for IBM's S/370 and S/390 mainframes. pSeries computers running the AIX operating system are IBM's third generation of UNIX computer platforms. (The IBM RT System was the first, and RS/6000 systems were the second generation.) Figure 3.6 shows a timeline of the evolution of AIX from its inception to the current version, AIX 5L v5.2. The chart lists the key functional enhancements of each version and indicates what was happening in the overall computing industry at the time. AIX 5L incorporates all the developments of previous versions that are still applicable in today's e-business world. Some functions have been dropped over the years, as the evolution of the computer industry made them obsolete.

As shown in Figure 3.6, the introduction of AIX 5L in April 2001 marked several key new capabilities for IBM's UNIX operating system. AIX 5L v5.1 provides a 64-bit operating system kernel and device drivers. It supports up to 32-way SMP servers like the pSeries 690, logical partitioning, up to 256 GB of memory, and Linux affinity tools. The second version, AIX 5L v5.2, offers the JFS2 file system, which allows file and file-system sizes of 16 TB and supports dynamic LPAR, dynamic processor and memory Capacity Upgrade on Demand, dynamic processor sparing with CUoD, and up to 512 GB of memory.

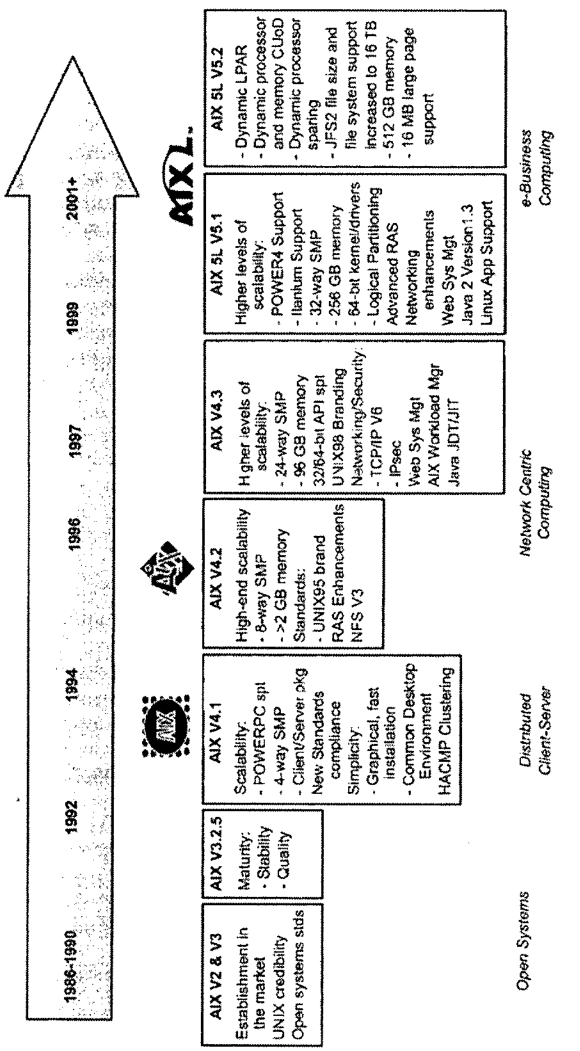


Figure 3.6. Timeline showing the evolution of AIX and the key features introduced with each version.

The AIX 5L operating system is a multitasking, multiuser, multiprocessor operating system adhering to industry standards. Figure 3.7 is a diagram showing the various functional elements that compose the operating system. This diagram is similar to the software structure described earlier in the chapter (see Figure 3.1, shown previously). User programs are still on the top, and system hardware is at the bottom. The distinction between "dynamic" and "static" components in the operating system kernel means that the dynamic components can be installed selectively, as requirements change or as technologies develop, without affecting the underlying static components of AIX 5L.

AIX 5L and its extensions perform all housekeeping tasks for pSeries systems and interact with users to do such things as starting application programs, changing passwords, erasing files, and so on. The command-driven user interface common to UNIX operating systems requires the user to type in somewhat cryptic UNIX commands and is generally used

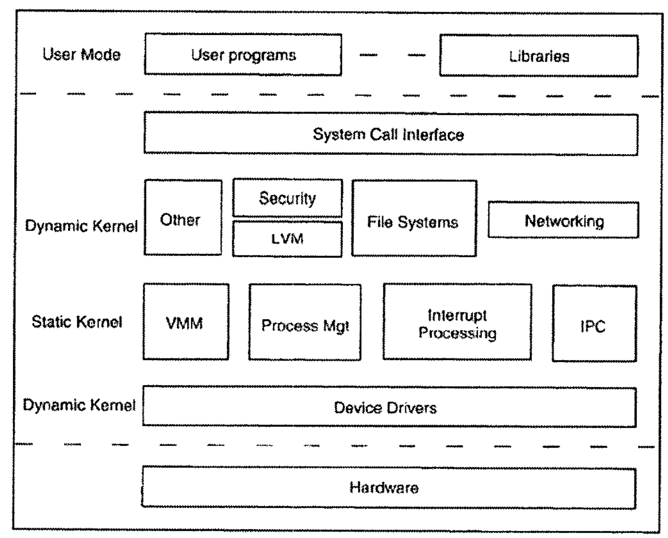


Figure 3.7. AIX 5L structure.

only by programming professionals. AIX 5L provides more user-friendly interfaces, including the Web-based System Manager, the AIXwindows7 graphical user interface (GUI), and the Systems Management Interface Tool (SMIT). These tools, along with on-line documentation, facilitate system administration and make the AIX 5L operating system easy to use. The AIX operating system maintains a high level of compatibility with industry standards to provide source-code compatibility with many application programs written for other open systems.

Traditional UNIX operating system application development tools (such as the Source Code Control System, or SCCS, subroutine libraries, and the Make command) are addressed, as well as some newer tools (e.g., object-oriented programming and computer-aided software engineering products). The communications functions in AIX 5L handle everything from traditional UNIX operating system communications (e.g., asynchronous ASCII protocol) to current industry standards (e.g., TCP/IP and the Network File System). As has been mentioned previously, in most pSeries installations, PCs are attached using local area networks or wide area networks (WANs) to allow users to monitor and control the system, share data, exchange information, and use shared devices such as printers.

To recap, IBM started with the UNIX operating system base, incorporated enhancements made by many organizations, and added some new ideas, all without losing compatibility with industry standards (POSIX, SVID2, or Open Group). The resulting AIX 5L operating system is an open system, which has evolved to conform to new industry standards as they emerge. One organization that has a major influence on the development of AIX 5L is the Open Group, a not-for-profit consortium of computer vendors (including IBM, HP, DEC, and others) formed in 1996 by the merger of X/Open and the Open Software Foundation (OSF). The mission of the Open Group is to develop specifications and software for the open-system arena and make the resulting specifications and software available to computer vendors under fair and equitable licensing terms. The Open Group solicits members and nonmembers for submissions of concepts and software to solve a particular problem, then evaluates each submission and selects the best. This submission then becomes the Open Group-endorsed standard for the open-system environment. This process is designed to foster the development of vendor-neutral, open-system standards that many different computer vendors can implement in a compatible fashion. IBM has committed to support Open Group technologies and standards as they emerge in AIX 5L.

A Closer Look at AIX 5L

The previous section provided an overview of the AIX 5L operating system, which has many complex features and functions, a complete description of which would warrant a separate book. The remainder of this chapter will look at some of the most important topics:

- User interfaces/services
- Systems management
- Disk management
- Application development
- Communications support

User Interfaces/Services

The manner in which a user interacts with a computer system is determined by the user interface provided by the program(s) being executed by the computer system. Like other things concerning computers, the style, ease of use, and productivity of user interfaces have evolved over time. The AIX 5L operating system offers several different user interfaces selectable by the user. These user interfaces can be broken down into two types:

- Command shells
- Graphical user interfaces

An example of a command shell is shown in Figure 3.8. This is a very simple user interface in which the user must type in AIX 5L operating system commands at a command prompt in order to accomplish operating system tasks such as starting application programs or examining the contents of a disk. For example, the user would type in "ls" (the list command) at the command prompt, as shown in Figure 3.9, to see a list of the files contained on disk.

Three different command shells are provided with the AIX 5L operating system, differing in the command syntax and prompt:

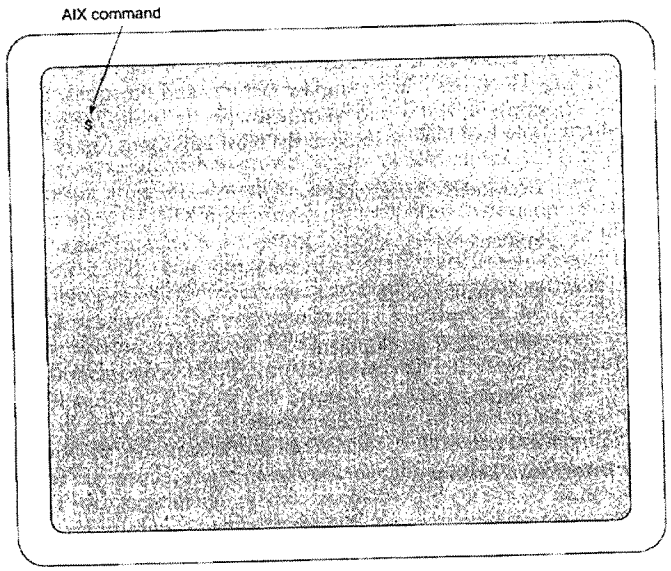


Figure 3.8. AIX command prompt presented by a command shell provided with AIX.

- Korn shell
- Bourne shell
- C shell

With any of these command shells, the user traditionally must remember (or look up) somewhat cryptic commands and type them in using the keyboard. For this reason, command shells are more useful to experienced users.

Of course, it is much easier—not to mention being expected as the norm—for users to interact with pSeries systems using a graphical user interface. Everyone is used to using a mouse and icons to initiate actions. The AIXwindows environment provides a graphical user interface. Figure 3.10 shows a typical image as presented by AIXwindows. By selecting various icons with a mouse, the user can quickly start an application program, browse existing files, create new files, and delete files. The many different styles and sizes of text available through the PostScript standard can be displayed on the screen, making for effective and attractive presentation of information to the user. This feature of AIXwindows, called Display PostScript, allows the user to view PostScript text and images on the screen. The Common Desktop Environment (CDE) is provided with AIXwindows and provides an icon-based interface for working with files and other basic tasks. Also included with AIX 5L as part of the Linux affinity tools are the GNOME and

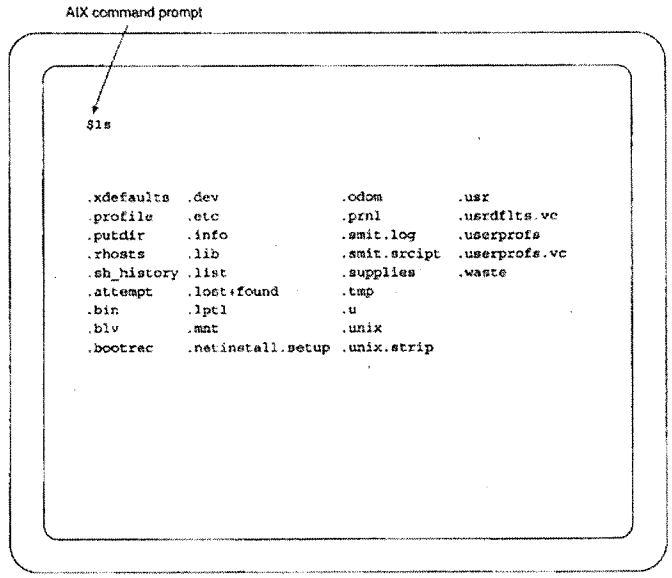


Figure 3.9. The "ls" command is used to list the files contained on a disk, as shown.

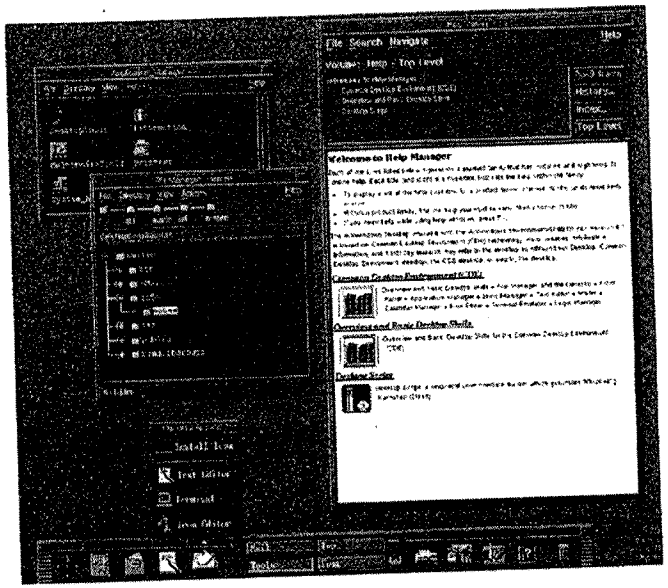


Figure 3.10. Graphical user interface provided by AIXwindows environment.

KDE interfaces familiar to Linux users. The user can select any of the CDE, GNOME, or KDE desktop environments.

In addition to the programming needed to perform traditional operating system tasks, AIX 5L includes some commonly needed programs that provide various services of interest to AIX users. The on-line documentation for AIX is available in HTML format and is searchable and viewable with a Web browser. The HTML documentation CD ships as a part of the base AIX operating system. InfoExplorer, which provides an interface to basic "help" information (Figure 3.10), continues to be supported as an optional AIX feature. More detailed technical reference documentation is available on-line in a Web browser-viewable format.

The standard UNIX mail facilities are also provided in AIX 5L. The user can choose from two different mail programs. The more basic mail program commonly found in today's UNIX operating systems allows users to create and send simple documents or messages to one another. For more advanced mail functions, users can choose to install the message handling program, which offers additional capabilities such as message sequencing, message annotation, mail folders, date sorting, and so forth.

Systems Management

With any computer system, various tasks must be performed in support of the computer system itself by someone trained to be a systems administrator. These are called systems management tasks and involve things such as authorizing new users and making backup copies of the disk information. These basic operating system tasks can be done by stepping through menus in the Systems Management Interface Tool of AIX 5L. More experienced administrators may use AIX 5L operating system commands from a command shell, as previously described.

In today's distributed world, this is no simple task, especially if one is managing a large number of machines that are geographically dispersed. For managing one or a small number of pSeries systems, the tools provided with AIX 5L are adequate. But for situations in which hundreds of systems and/or a heterogeneous mix of pSeries and other systems must be managed, IBM offers the Tivoli 7 suite of systems management products, which form the basic infrastructure to link together a vast array of systems from various suppliers and running a variety of operating systems.

One of the first jobs to be done by a systems administrator is to tell the system who is authorized to use it. This is done by creating a user profile for each system user. The user profile contains information such as the user's nickname (called a user identification, or user ID), password, and security level, and accounting information for departmental billing for computer services. The AIX 5L operating system's built-in security, if enabled, will require the user to enter the correct password before being allowed access to the system. If desired, security can be defined so that the user is restricted to specific functions. In fact, security enhancements were made to AIX 5L based on the stringent security

requirements of the National Computer Security Center Trusted Computer System Evaluation Criteria Class C2. Several useful security enhancements in AIX 5L involving password management and login controls are covered in more detail at the end of the chapter.

Now that the users can begin to use the pSeries system, the disks will begin to accumulate information that may be vital to the day-to-day operations of the organization. This information becomes an asset to the organization and should be protected as such. AIX 5L provides several functions that allow the systems administrator to protect against the loss of this information, be it from user errors, hardware failures, intentional corruption, or whatever. Through AIX 5L operating system facilities, the systems administrator can make backup copies of the information on disks to magnetic tape. These backups should be done on a regular basis (e.g., daily), and the backup tapes are stored in a safe place at another physical location. If the information on the system disks is somehow lost, the backup tape can be used to restore the system to the state when the last backup was made. Several programs familiar to UNIX administrators (tar, cpio, and backup) are available within AIX 5L or as additional products (Sysback and Tivoli Storage Manager) to facilitate backing up the system.

In environments in which quick recovery from disk failures is even more important, a function of AIX 5L called mirroring allows for duplicate copies of information to be kept current on two or three different disk drives. This function allows for immediate recovery from disk failures. We will cover mirroring further in the next section.

Because pSeries systems are often used in environments in which operating system experience is limited, the AIX 5L operating system provides assistance in this area. SMIT provides menus to guide the systems administrator through administrative functions such as creating a user profile, adding a new printer, managing disk storage, or changing a password. Rather than having to remember and type in the somewhat cryptic commands, the user can select menu items and will be prompted for needed information. Then AIX 5L automatically builds, issues, and logs the command(s) needed to accomplish the task. While the user is interacting with SMIT, on-line help text is available to help resolve any confusion. More experienced systems administrators can still choose to directly issue traditional commands to accomplish these same tasks. Figure 3.11 shows a typical screen using the Web-based System Manager. This interface allows the systems administrator to monitor and control one or more systems either from a graphics terminal attached to the

pSeries system or on the Internet from a PC with a Java 1.3-capable Web browser. The names of the host system(s) to be managed are shown in the navigation area at the left, and the contents area at the right shows the items that the Web-based System Manager can handle for that host. As can be seen in Figure 3.11, rather than using terminology that is unique to UNIX, the Web-based System Manager uses concepts and symbols that are familiar to users experienced in managing systems using the Microsoft Windows 7 operating system.

Many customers take advantage of ordering pSeries systems with AIX already installed on the disks at the factory. The installation can be customized to include only those components of AIX 5L that the customer will use. For an additional fee, IBM manufacturing offers a service of installing other software and configuring pSeries systems to the customer's specifications. Some IBM Business Partners may offer installation and customization services for pSeries systems purchased from

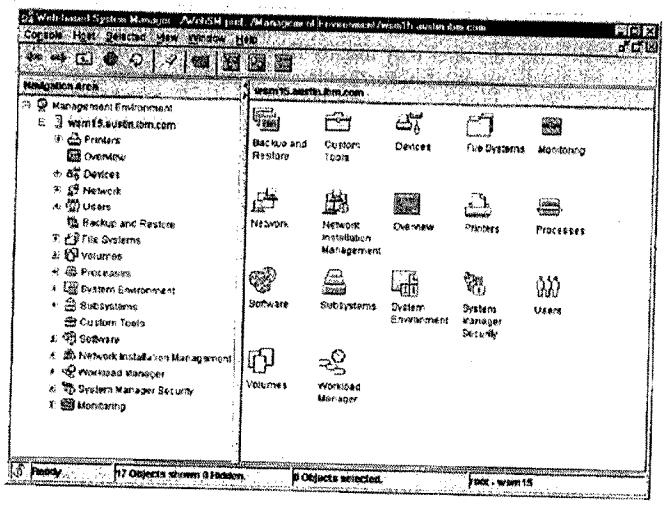


Figure 3.11. A Web-based System Manager screen showing the monitoring and control tasks which the systems administrator can select.

them. While the installation of AIX 5L (at no extra charge) is used by many customers, the custom system configuration service is usually of interest only to customers who are buying multiple pSeries systems, all of which must be configured identically. An example of this type of customer is a large retailer who must deploy many pSeries systems to stores throughout the nation or throughout the world.

An additional software package called the Service Agent is available at no additional charge if your pSeries system is covered by an IBM Warranty or IBM Maintenance Service Agreement. The Service Agent for pSeries does the following:

- Monitors and analyzes all recoverable system errors, and, if needed, can automatically place a service call to IBM without user intervention.

- Performs problem analysis on a subset of hardware-related problems and can automatically report the results to IBM Service with customer authorization (and a modem). The Service Agent analysis and problem reporting is based on default values, but the customer may modify those values. This may be important during periods when hardware upgrades or testing are in progress in order to prevent placing an unnecessary service call.

With Service Agent, it is possible for IBM Service to show up to perform preventive maintenance that the user had no idea was needed. In this way catastrophic failures can be avoided. Alternatively, the customer may configure Service Agent to alert its own IT personnel by e-mail. This can be done in lieu of or in combination with placing a call to the IBM Service Center.

Disk Management

One of the jobs performed by the systems administrator of a computer system is to subdivide and allocate (manage) the available disk storage to meet the needs of the users. The systems administrator uses functions of the operating system to manage the disks. Many UNIX operating systems conceptually subdivide a computer's disk storage into many smaller disks called file systems. These file systems can then be used to hold programs, data files, and so on. The systems administrator must

decide how large each file system needs to be during the initial installation of the application program. However, it is often difficult to judge how large a given file system needs to be until the system has been in productive use for a while. Because it is difficult to increase the size of a file system after the system is in productive use, herein lies a problem with the disk management approach taken by most traditional UNIX operating systems. Another limitation commonly found in UNIX operating systems is that a single file system cannot span more than one physical disk drive. That is, the file system size is limited by the size of each individual disk drive installed in the computer system rather than by the total amount of disk storage in the computer system. As application programs grow more sophisticated, file systems larger than a single physical drive are desirable.

To address these limitations, the AIX 5L operating system has expanded on the file system concept of the UNIX operating systems (and earlier AIX versions) with the Logical Volume Manager (LVM). The LVM conceptually partitions a group of physical disk drives (up to 128) into equal-sized sections called physical partitions. A logical volume is a collection of these physical partitions, conceptually equivalent to a file system. One key difference in this approach is that a single logical volume can consist of physical partitions from multiple disk units in the computer system. This means that a logical volume can be larger than any individual disk unit in the system if necessary. Another difference is that physical partitions can easily be added to a logical volume to increase its size without disrupting normal operation or moving any other partition.

One other function of the LVM is mirroring. This feature allows for multiple (or mirror) copies of a logical volume to be automatically maintained in disk storage. Each physical partition of a mirrored logical volume has one or two other physical partitions allocated on different physical disks to hold identical copies of the data. If a permanent disk error occurs while reading a mirrored logical volume, the LVM will automatically read the data from one of the mirror physical partitions and write any new data to a new area of disk. Thus the mirroring function of the LVM often allows a pSeries system to recover from permanent disk errors without disrupting normal system operation. Systems providing redundancy for automatic error-recovery purposes are said to be designed for high availability. Be aware, however, that because you are keeping multiple copies of information in the disk system, more disk space will be required when the mirroring function is used.

The JFS2 file system introduced in AIX 5L has been tested to support file sizes up to 16 terabytes, with an architectural limit of 4 petabytes (4 PB = 4000 TB). It also uses a B-tree directory structure that provides improved performance. JFS2 supports on-line disk defragmentation and software level striping and mirroring (RAID 0 and RAID 1).

AIX 5L v5.2 provides "large page" support. That is, software may define pages as large as 16 MB to help improve throughput for computation-intensive workloads that require large amounts of data to be transferred to and from disk storage.

Application Development

Many organizations find that writing their own custom application programs is the best method of solving problems through computers. We will discuss various features within AIX 5L and additional products that can be used to develop applications. A large number of development tools are available for AIX; after all, that's where UNIX began, as a development system. First, let's look at what comes with AIX 5L itself.

AIX 5L Development Tools

With the AIX 5L operating system, several development tools are available to you. AIX 5L comes with the "vi" full-screen editor and tools to support the development and maintenance of custom application programs. These include traditional UNIX commands and utilities such as the Make command, used to easily rebuild complex systems of programs after changes have been made. Another traditional UNIX operating system function supported by AIX 5L is the Source Code Control System, which provides a mechanism to record and control when, why, and by whom changes are made to the source code of a program under development. Subroutine libraries provided include libc.a, the enhanced floating-point math library, the 4.3 BSD compatibility library, and a DCE threads compatibility library.

AIX 5L also provides a type of programming structure for automating a series of operating system commands, called "shell scripts." Shell scripts can be used to issue a single operating system command or to define a complex series of operating system commands that present the user with menus and initiate other application programs. No matter how many operating system commands are contained in a shell script, all commands in the shell script can be executed by simply typing in the name of the shell script. Shell scripts are a shorthand for users or systems administrators, allowing them to automatically execute long lists of operating system commands.

When the graphical Common Desktop Environment is used, several additional development tools are included, such as a graphical language-sensitive editor, a program build manager, static analysis tools, development and tool managers, a graphical debugger, integrated file transfer, and so forth.

Compilers

The IBM VisualAge C Compiler, a separately ordered product for AIX 5L, supports the development of programs written in the popular C programming language. A programming language can be thought of as a library of instructions from which a programmer constructs a program. The instructions of C (and other programming languages) are designed to be easily understood by the programmer, facilitating productivity while writing programs. However, the resulting list of instructions (the program) cannot be directly understood by the pSeries hardware. The VisualAge C Compiler takes the list of C instructions written by the programmer and automatically converts them to a series of instructions that are executable by the pSeries processor circuitry.

Other languages available include C++ and FORTRAN. Additional compilers for AIX 5L are also available from companies other than IBM. All of these compilers are designed to meet existing industry standards that specifically define the elements of each language. The language you select depends on the requirements of the application program and the skills of your programmer(s).

Although not a programming tool in the same sense as are compilers, it is important to note that AIX 5L provides runtime execution for programs written in the popular industry-standard Java 2 language. One may think of this capability as a Java 2 language interpreter. Both the 64-bit and the 32-bit kernels of AIX 5L are fully Java 2 compatible.

Libraries

Several subroutine libraries are offered for the AIX 5L programming environment, including the AIX Engineering Scientific Subroutine Library (ESSL). These subroutines use algorithms tailored to specific op-

erational characteristics of the hardware such as cache size, page size, and floating-point operations. ESSL provides prewritten routines for such things as matrix operations, eigensystem analysis, interpolation, linear algebraic equations, Fourier transforms, convolutions and correlations, and others. This subroutine library is designed to accelerate application program development by allowing the programmer to simply embed highly tuned, prewritten mathematical routines rather than write them from scratch.

Through the X development environment, AIXwindows (discussed earlier in the chapter) provides a multiwindow (based on OSF/Motif) programming environment. This allows an application programmer to do things such as examine source code in one window and monitor the executing program in another. Alternately, programmers may be accessing AIX 5L on-line documentation in one window while writing a program in another, looking at both simultaneously. AIXwindows also provides tools to ease the development of effective graphical user interfaces for programs—namely, the X Windows Graphics Support Library and Display PostScript for AIXwindows product.

Support for three-dimensional (3D) graphics is provided by two software components for use with IBM's UNIX workstations—OpenGL and GL 3.2, and PHIGS. The OpenGL and GL 3.2 component provides support for the popular Graphics Library developed by Silicon Graphics. The PHIGS component provides a set of subroutines for generating and manipulating graphical images through the ANSI and ISO standard known as the Programmer Hierarchical Interactive Graphics System (PHIGS). Programs specially written to these APIs can usually be migrated to AIXwindows with minimal programming efforts.

Additional Tools

Many of today's larger program development projects have become extremely complex—involving tens, hundreds, or even thousands of people who may be geographically dispersed around the world. Managing such projects and ensuring a quality result has become a discipline in and of itself, known as computer-aided software engineering (CASE). Although CASE methods were developed to address the needs of large application development projects, CASE concepts and tools can be employed for any size project. Today, many computer-based CASE tools are emerging to help implement a structured application development process. In fact, there are so many different tools and approaches

intended to improve the process of custom application development that the diversity itself can lead to incompatibilities and a fragmented development process. This problem has been rectified by the Common Desktop Environment. The CDE provides a common framework for many CASE tools to work coherently together. This open approach allows the CASE tool user to choose from a wide array of CASE tools (that adhere to the framework) from many different companies with the confidence that they will work together. Many of the standard AIX 5L development tools are integrated via the CDE.

A product similar to the standard SCCS called Configuration Management Version Control (CMVC) can be especially important in larger projects, where multiple programmers may be working on a single project. CMVC can handle the check-in, check-out, versioning, and reporting procedures on any number of development items such as source code, object code, documentation, and design changes.

The built-in features of the AIX 5L operating system along with additional tools (AIXwindows, optional compilers, the CASE framework/tools, and so on) make for productive and flexible application program development and maintenance.

Distributed Computing

Among the most far-reaching extensions to AIX 5L are the AIX Distributed Computing Environment products. In the open-system environment, many different brands of computers are typically found in the same communications network. An environment in which many small computers (rather than one large central computer) are used to satisfy the computing needs of users is called a distributed computing environment. Although the computer systems are physically attached via the network, often the application program(s) running on one computer system or operating system can't interact effectively with an application program running on a different computer system or operating system—even when no one is breaking open-system rules. This problem becomes more apparent as networks grow and users on one workstation (a client workstation) need access to the resources located on another computer (a server).

One way to address the need for a more homogeneous environment for distributed computing is to provide a set of common services that are always present in a distributed computing environment. Then any

application programs that are designed to use only those services will work on any type of computer system and network. The Open Group (discussed earlier in this chapter in "AIX 5L—An Executive Overview") has selected and endorsed a set of distributed computing technologies taken from many different vendors and collectively called them the distributed computing environment. IBM's AIX DCE products are built on the Open Group's DCE.

Many of the base "run-time" services of DCE and its major service, the Distributed File System (DFS), are integrated into AIX 5L. These client services include configuration, security, cell directory, time, RPC, and DFS. The DFS base services allow users to share files stored in a network of computers without knowing the physical location of the files.

Communications Support

Communications facilities included in AIX 5L allow a properly equipped pSeries system to communicate with almost any kind of computer in a variety of ways. Traditional open systems used terminals and printers that communicated with the computer system over asynchronous communications links using the American Standard Code for Information Interchange protocol. With this communications method, one character at a time (letter, number, period, comma, and so on) was sent from the computer's communications port through a simple cable to the communications port of the attached device. This is how all ASCII terminals and printers communicated with the UNIX operating system. This simple ASCII method was also used to send information from one computer system to another over a similar type of cable. Although pSeries systems and AIX 5L still support ASCII communications with other computer systems, they are much less frequently used than the methods described here.

As more and more vendors offered computer systems and the need for intercomputer communications evolved, the Transmission Control Protocol/Internet Protocol (TCP/IP) was developed. As an industry standard, TCP/IP allows systems from many different vendors to communicate with each other in a single network. The AIX 5L operating system supports TCP/IP over Ethernet/802.3 networks, Token-Ring networks, the Enterprise Systems Connection (ESCON7), Asynchronous Transfer Mode (ATM), and the Block Multiplexer Channel Adapter if the computer systems are near each other (for example, on the same campus). AIX 5L also supports the TCP/IP protocol between distant computer

systems—over asynchronous communications networks or X.25 packet-switching networks, for example.

Communications functions have emerged to meet the needs of more sophisticated communications environments, such as the Network File System (NFS) and the Distributed Computing Environment, both of which are supported in the AIX 5L operating system. The NFS and application programs written to use it allow one computer system to access information located on another computer system over a local area network or other networks using TCP/IP. This allows information to be distributed in a network so as to make that information most available to those who need it and so that it can best be managed. The DCE extensions to AIX 5L support application programs designed to take advantage of DCE by distributing a computer's workload across the various computer systems in a network. However, with recent advances in communications technology, DCE is of less importance today than in the past.

Although the requirement for pSeries users to access applications on other types of systems in native mode (e.g., emulating an IBM 3270 or 5250 terminal) is rapidly going away, AIX 5L supports the IBM Host Access Client Package for Multiplatforms v3.0 product. The IBM Host Access Client Package provides access to applications and data on iSeries (5250), zSeries7 (3270), and DEC/UNIX (VT) hosts for traditional and Web users in either intranet or SNA environments. AIX 5L is supported both as a client and as a host.

Tivoli NetView7 provides network management tools based on the Simple Network Management Protocol (SNMP) commonly used in the open-system environment. It works like this: When a device in the network (called the SNMP agent) senses it has a problem, it sends an SNMP message to the computer system running NetView (the SNMP manager). NetView then notifies the network support personnel that a problem has been detected and provides detailed information about the problem. Thus, NetView allows for centralized management of a distributed, open-system network, which typically improves the productivity of the network support personnel and the availability of the network. Tivoli NetView also provides firewall support with a single unidirectional port for use in e-business networks and provides the systems administrator many tools for managing the network, including diagnostic aids and a graphical representation of the network.

An optional, layered product for AIX 5L is AIX Fast Connect for POWER Version 3.1, which allows a pSeries system to cooperate with personal computers. AIX Fast Connect provides file and print services

for connecting many types of client workstations, such as Windows 9X, Windows NT client, Windows 2000 client, Macintosh, and NetWare clients. These components all use the same record locking, file locking, and security functions, enabling users to share files between AIX Connections, NWserver, and MAC Server and AIX applications, and to transparently and concurrently share files with each other's clients. The IPX/SPX, TCP/IP, NetBIOS, RFC 1001/1002, and AppleTalk protocols are supported, as well as an embedded gateway support for NetWare, server message block (SMB), and DCE-DFS file systems.

AIX Fast Connect provides for the free exchange of information between the PC and the pSeries system. For example, information in pSeries files can be brought down to the PC and manipulated using a spreadsheet program. Further, a file created on the PC (using a word processor, for example) can be transferred up to the pSeries system and accessed by AIX 5L like any other file. Finally, the pSeries disks can be used as an extension to the PC's disk space through the use of shared volumes—areas of the disk that can be accessed by either a PC or AIX 5L. Through these shared volumes, the pSeries system can be the central storage element in a network of PC and pSeries system users, allowing any authorized user access to the information. In the same way that information can be shared, printers and communications links can also be shared. For example, through the AIX Fast Connect virtual printer function, a PS/2 user can automatically send documents to printers attached to a pSeries system.

AIX Fast Connect allows the PC and the pSeries system to cooperate in such a way that the user can take advantage of the best of both. The PC can be used for such operations as spreadsheet programs, graphics, and word processing, providing functions and response times tuned to these applications. At the same time, the pSeries system contributes information security, centralized systems management, large disks, extensive communications, high-speed printers, and so forth. AIX Fast Connect allows the user to perform the needed functions on the system best suited for the job.

Internet Software

The explosion of the World Wide Web has caused an outpouring of software for the Internet from vendors. IBM is no exception, and in fact much of its Internet-related software for AIX 5L is included as a free Expansion Pack. In general, the software falls into a few categories: browsers, server enablers, and development tools. For more information on the Internet and its business possibilities, read *Dr. Livingstone's Online Shopping Safari Guidebook*, by Frank Fiore (Maximum Press), *Marketing on the Internet*, by Michael Mathiesen (Maximum Press), or you can check out the Maximum Press home page at *http://www.maxpress.com*.

Web Browsers

Browsers are basic enablers that allow any client machine to search for, retrieve, and navigate (browse) HTML (HyperText Markup Language) data. The AIX 5L Expansion Pack for AIX 5L includes the Netscape Communicator Web Browser, which allows Web pages to be accessed from the Internet and viewed locally on the client desktop. The Netscape Navigator built-in scripting language, called JavaScript, is supported in this version of the Netscape Navigator and is what allows much of the moving elements that you see. JavaScript extends and enhances the capabilities of HTML documents. A Mail Window allows you to read your e-mail, sort the messages, file them in folders, and mail replies. The News Reader provides a concise view of the news hosts, newsgroups, and messages. The address book allows organization of e-mail addresses, and it looks and works very much like the bookmarks. Support for progressive JPEG images gives high-quality images that load faster than regular JPEG or GIF images.

For easier browsing of Internet documents, the AIX 5L Bonus Pack also includes the Adobe Acrobat Reader. The Acrobat Reader is part of the Adobe Acrobat family of software, which lets you view, distribute, print, and save documents in Portable Document Format (PDF) regardless of the computer, operating system, fonts, or application used to create the original file. Portable Document Format files retain all the formatting, fonts, and graphics of the original document, and virtually any PostScript document can be converted into a PDF file. The Acrobat Reader software package includes the Acrobat Reader program file, associated files, and several on-line documents.

Web Server Enablers

These enablers allow businesses that have information to be shared either within the company (intranet) or with the outside world (Internet) to use a pSeries system as the server of that information. In fact, pSeries

systems have proven themselves to be quite good at Web serving. For example, at the time of this edition, the 16-way pSeries 690 result of 21,000 and the 8-way pSeries 650 result of 12,400 are numbers one and six, respectively, on the Ideas International list of top performers on the SPECweb99 benchmark, which measures the number of Internet connections that can be handled per second. Check the Maximum Press home page at *http://www.maxpress.com* for the latest information.

AIX 5L includes the IBM HTTP Server, a Web server based on the popular Apache Web Server. It is also used in the Web-based System Manager and to access Help on-line. It supports such security features as SSL (Secure Socket Layer) connections and hardware cryptography. The WebSphere Application Server, a separate program product, is designed for nonprogrammers to use at home, at work, or in a small business. If you can use the Netscape Navigator browser, then you can publish your own Web site with WebSphere Application Server. It is a user-friendly, entry-level Web application server that installs quickly.

Development Tools

The AIX 5L Bonus Pack and Web Download pack include the IBM AIX Developer Kit, Java 2 Technology Edition, Version 1.3.1. This software package supports the Java programming environment. Java is an object-oriented programming environment that operates independent of any operating system or microprocessor. Java programs, called applications or applets, can be entirely developed using the compiler, debugger, and applet viewer tools provided in IBM's implementation for the AIX for Java development environment. (C and C++ compilers and tools are not needed to create/run Java-based applets.) The same applets can be dynamically transmitted over a network and run on any client that has been enabled for Java.

Because applet execution is platform independent, an applet developed with the AIX 5L tools can be executed on any Java-enabled platform (for example, Solaris). Typically, end users run Java applets from a Java-enabled Web browser (such as Netscape's Navigator). Clicking on a link in a Web page results in an automatic download of the applet over the Internet to the client machine, with subsequent execution of the applet on that machine. Native Java-based security technology is used to enhance the secure execution of Java applets.

Applets enable World Wide Web users to deliver more visually compelling Web content, such as through the use of animation. For example, the end user would potentially be able to view and interact with an applet by rotating a product image dynamically. Applets can also be created to take advantage of native tools on specific platforms. For example, an applet could be written to specifically interact with a database running on AIX 5L to dynamically extract and post live information on a user's graphics display such as live weather maps or 10-minute-delayed stock prices. Applets can be discarded easily after they have been run, or they can be saved to media for future use. A set of sample applets is included to demonstrate a number of features of Java.

AIX Workload Manager

AIX Workload Manager (WLM) is a standard feature of AIX 5L and is intended to help system administrators apply resource management policies to different workloads and help manage peak system demands. This capability can also be used to consolidate workloads from several different systems onto one pSeries system.

The WLM function provides a policy-based method for managing system workload and system resources, including the following:

- System resource allocation with fine granularity to specific jobs
- Logical job separation on the server
- Keeping applications in memory for more predictable performance
- Dynamically changing policies as job requirements change
- Control options that include minimum and maximum percentage limits, shares, or a combination of both

As a function of AIX 5L, WLM will operate on any pSeries system from the smallest to the largest. WLM can also be used in an LPAR environment. Within each logical partition, the system resources assigned to that LPAR can be allocated to certain applications, just as if the LPAR were an independent server. This feature is extremely useful in guaranteeing proportional or minimal levels of resource to various applications.

AIX WLM is a very important tool used by the system administrator to allocate system resources (processors, memory, I/O) to processes and applications, so that the specified applications will always have available the resources needed to maintain adequate performance. In other words, with WLM, the administrator can prevent certain applications from being "starved" for resources due to peak demands from other application workloads.

WLM allows management of subsets of workloads and control of subsets of total system resources. In addition to a graphical user interface, new functions available to the system administration are the ability to allocate disk I/O bandwidth and a subsystem to perform resource usage accounting per workload class, in addition to accounting per user or group of users.

Performance Analysis Tools

The IBM Performance Toolbox (PTX) and AIX Performance Aide v3.1 provide a complete set of tools for analyzing and presenting in visual form the performance of the system resources, both in local and distributed environments. When used with AIX 5L v5.2, these tools can monitor resources that are dynamically assigned to LPARs. PTX v3.1 includes an application that presents simple snapshot views of overall system performance. With Performance Aide v3.1, the system administrator can continuously monitor the top resource consumption of processors, memory, and disk storage. These tools provide the detailed data that the system administrator needs in order to use AIX Workload Manager, dynamic LPAR, and pSeries Capacity Upgrade on Demand features in the most cost-effective way.

Networking Performance

In today's "connected world" in which users access a server via the Internet and in which multiple servers within an enterprise are networked together, AIX 5L provides many capabilities to help ensure the best possible network performance. The administrator has the option of configuring multiple routes for load balancing or of setting up alternate routes for network traffic when the best route cannot perform its task for some reason. Using the Virtual IP Address (VIPA) function, system administrators can define a virtual IP address for a host, thus decoupling the IP address from a physical interface for TCP connections. A Dynamic Feedback Protocol (DFP) provides load statistics so that load can be balanced by sending future connections to available servers. To help improve TCP/IP performance over congested networks, AIX 5L provides increased initial windows, explicit congestion notification, and limited transmission mechanism functions. Sendmail Version 8.11 improves performance with multiple queues, memory-buffered pseudo-files, and more control over resolver timeouts. Also, AIX 5L supports the Cisco EtherChannel technology, which allows aggregation of up to four Gigabit Ethernet ports into a single channel. These network technology functions may seem a bit esoteric and of little interest to the average person, but for the systems administrator responsible for a company's e-business network, these tools and programming interfaces provided in AIX 5L can make a significant difference in the system responsiveness that will be perceived by the end users of the system.

Linux Affinity

AIX provides the AIX Toolbox for Linux Applications, packaged on separate media for the convenience of users who may not need to take advantage of this package. This toolbox contains a collection of open-system software built for AIX with APIs to AIX so that recompiled Linux applications using these routines do not have to supply their own libraries. IBM does not support these open-system tools, but they have been tested on AIX 5L. So, for those customers who want to port applications written for Linux to run as AIX 5L applications on their pSeries server, IBM has provided the tools to make that as easy as possible. Alternatively, users may choose to run Linux in one or more partitions of pSeries servers that support LPAR or may choose to run Linux only (i.e., "natively") on their pSeries system.

Other AIX Features

When AIX 5L is preinstalled, a pSeries Welcome Center is optionally available at no extra charge. The pSeries Welcome Center is a user-friendly introduction to the pSeries and AIX 5L product families that provides information about the end-user's pSeries system and software, facilitates quick setup and configuration, demonstrates system capabilities, and provides links to other valuable resources of pSeries and AIX

information such as service and support. The Welcome Center uses the Netscape Communicator as the presentation tool with a default "home page" and a library of local HTML-format documents. These take up about 20 MB of local disk space, but, after viewing, there are options to remove them. The Welcome Center covers four major topics:

- "About Your System" familiarizes the end user with the features of the pSeries and AIX.
- "System Setup and Registration" facilitates quick setup and system configuration.
- "Just Imagine" contains a variety of topics including a demonstration of multimedia capabilities.
- "Contact IBM" provides quick access to the service and support organization.

From a mail server standpoint, AIX includes support for the industry-standard Post Office Protocol III (POP III) and Internet Mail Access Protocol 4 (IMAP4).

With AIX 5L, it is possible to boot a pSeries server from a storage area network (SAN) disk, so the systems administrator can manage and control the boot image from a central storage subsystem. There are other tools in AIX 5L to help the systems administrator manage multiple systems in a distributed system framework. Reliable Scalable Cluster Technology allows the administrator to monitor system resources and optionally run automatic responses if certain thresholds are met, and the Web-based System Manager provides a management console capable of managing multiple hosts without a Web browser. For users of pSeries and RS/6000 systems with the GXT4500P and GXT6500P 3D graphics accelerators, OpenGL applications can now support 64-bit direct window access, which should increase performance of 64-bit OpenGL applications.

AIX 5L Bonus Pack, Expansion Pack, and Web Download Pack

When AIX 5L media is ordered, IBM provides at no additional charge two CD-ROMs with additional software offerings. The AIX 5L Bonus Pack and Web Download Pack offer a large number of additional tools

and applications, including evaluation software. The AIX 5L Expansion Pack provides AIX extensions, such as the IBM HTTP Server, Netscape Communicator, and Data Encryption Standard Library Routines. These offerings are provided mainly for users who need to take advantage of the Internet. For instance, there are Java development tools, security and cryptographic tools, network management tools, and the software for viewing AIX 5L documentation on-line. Links to the IBM Web site for a complete, up-to-date listing of the contents of the Bonus Pack, Expansion Pack, and Web Download Pack are available on the companion Web site for this book *(www.maxpress.com)*.

The Bonus Pack contains both ready-to-use applications and evaluation software packages. Over time, IBM plans to phase out the Bonus Pack on CD-ROM and provide these programs for users via the Web Download Pack. At this time, the following are some of the Bonus Pack contents:

- Adobe Acrobat Reader 5.06
- AIX Developer Kit, Java 2 Technology Edition, Version 1.3.1, both 32-bit and 64-bit versions
- AIX Fast Connect V3.1.2, Evaluation Software
- Geodesic Systems Great Circle Version 6.0.1.5, Evaluation Software
- Modular I/O Library V2.3
- Tivoli Storage Manager v5.1, Evaluation Software

The contents of the AIX 5L Expansion Pack are primarily security and encryption tools. Since many of these programs are not suitable for downloading on the Internet, it is anticipated that the Expansion Pack will continue to be provided on CD-ROM. At this time, some of its contents include the following:

- IBM HTTP Server v1.3.19.4
- Certificate Authentication Service v5.2
- Crytographic Library v5.2

- Network Authentication Service v1.2.0.1
- IBM Web-based System Manager Security v5.2
- Tools to Build Secure Java Applications v1.3.1
- Netscape Communicator 4.8 with 128-bit encryption

The Web Download Pack requires registration, which is free, for first-time users. Rather than being a direct repository for the software (as are the CD-ROMs), the Web Download Pack contains information about the offerings and pointers to the actual download sites—whether IBM sites or those of third-party software vendors. The Web Download Pack is more flexible because users do not need to have the CD-ROM at hand. Also, its offerings will be more up-to-date since new software and updated programs will be available right away, without having to wait for the next release of a CD-ROM.

Open-System Standards

AIX 5L is UNIX 98 compliant and conforms to standards for compatibility with other UNIX systems, including support for a range of standards and specifications including POSIX 1003.4a Draft 7 and X/Open XPG4 Base Profile.

In addition, AIX 5L meets the common application interface defined by the Single UNIX Specification (formerly Spec 1170). Spec 1170 is a common API specification that is a collection of the most popular and most widely used UNIX application programming interfaces (XPG4, SVID3, OSF, and so on). Many vendors are expected to implement this specification, making UNIX application programs more portable. As a result of meeting the specification, AIX 5L has been granted the UNIX brand from X/Open (now the Open Group) as discussed earlier.

Security Enhancements

AIX 5L has obtained an Information Technology Security Evaluation Criteria (ITSEC) F-C/2/E3 and an International Computer Security Association (ICSA) Virtual Private Network (VPN) evaluation. This is a lot of technospeak to say that AIX 5L has been certified as very secure by organizations that know security. Two of the most used and most

useful security enhancements are password management and login control. Password management is designed to do the following:

- Define and restrict words that cannot be chosen as passwords
- Set password expiration warnings
- Restrict the reuse of previous passwords
- Lock out users after a password exceeds a specified lifetime

The second security enhancement provided is login control, which provides the following features:

- Time of day/day of week restriction at the user level
- Time of day/day of week restriction at the login port level
- Lock out user after a specified number of failed login attempts
- Disable login port after specified number of failed login attempts
- Login retry delay
- Login completion time-out

Desktop User Interface

AIX 5L includes the industry-standard graphical user interface called the Common Desktop Environment. This is a Motif-based user interface based on the joint development efforts of IBM, Hewlett-Packard, Sunsoft, and others. The CDE is an easier to use, iconic interface for the graphics system user. It includes productivity tools such as a calendar system, e-mail, and an editor, clock, file manager, icon painter, and trash can. In addition, AIX 5L includes the GNOME and KDE desktop environments familiar to Linux users.

File System

The JFS2 file system in AIX 5L v5.1 supports both file systems and individual files up to 1 TB in size, and JFS2 in AIX 5L v5.2 provides

files and file systems up to 16 TB in size. AIX 5L provides support for disk defragmentation and software level striping. The software striping is similar to RAID 0 (discussed in Chapter 2) and allows parallel reading or writing of sections of data across multiple disks. Striping can enhance the performance of many applications. Software striping and mirroring can be combined on the same logical volume, providing a RAID 0+1 implementation via software.

Installation

If you don't have AIX 5L preloaded onto your system, tools to allow easy installation are provided. You can use the Web-based System Manager, which allows access from PC clients running Web browsers (e.g., Netscape Navigator, Microsoft™ Internet Explorer) that accept the Java plug-in. Other alternatives are a guided tour through the installation process, called Installation Assistant or the Network Install Manager (NIM), which installs the operating system from a server onto clients on the network. AIX 5L can install many of its components automatically as it detects the types of hardware options installed. It also has a deinstallation facility.

What About 64-Bit Capability?

Before we go into further detail, let's discuss this whole notion of 64-bit computing capability and what it means to you. This is an issue that has been confusing to many people for a number of years. After all, not too long ago we were all running operating systems with just 16-bit addressing on our desktops.

The primary advantage to using more bits is in gaining addressability for more memory. Until very recently, most UNIX operating systems were 32-bit systems, allowing memory up to 4 GB to be addressed. Now, 4 GB is an awful lot of real memory, but the performance of very large applications can be improved by having even more. When you go to 64-bit addressability, the theoretical limit on your memory goes up to a whopping 16 million TB (16,000,000,000,000,000,000 bytes). Figure 3.12 illustrates the exponential growth in addressability as the number of bits available grows. It should be several years before we outgrow the capabilities in 64-bit addressing (notice, we don't say never).

Number of Bits	Addressable Bytes	Theoretical Memory Limit
8	2 raised to the power 8	256 bytes
16	2 raised to the power 16	64 KB
32	2 raised to the power 32	4 GB
64	2 raised to the power 64	16 million TB

Figure 3.12. Memory addressability for operating systems with varying numbers of bits.

Let's assume that you are a manufacturing firm and that you have a relatively large database of, say, 30 GB, which you make available to your enterprise applications such as human resources, accounting, and distribution. If you have a UNIX system with 16 GB of main memory, you can load very large segments of your database into real memory, increasing the likelihood that the data you require will be in main memory. This allows you to avoid the performance hit of going to the disks for the information, swapping or paging this data into real memory, and writing the replaced data back out to the disk.

The preceding example assumed a 64-bit database as well as operating system and hardware, thus allowing big "chunks" (more than 4 GB) to be manipulated at one time. However, even if an individual database is not that large, performance advantages can still be realized by having a 64-bit operating system and larger memory available. In this case, you may have several applications and many processes running concurrently on your system. Even though no one of them wants more than, say, 2 GB or 3 GB of memory (or even considerably less), greater amounts of total memory will reduce memory contention between the processes that are running.

AIX 5L gives the application developer and the end user a full 64-bit environment. It supports 64-bit hardware, which includes all pSeries systems and currently available RS/6000 systems except Models B50 and 150, and 64-bit applications, such as Oracle 8 and IBM DB2. AIX 5L has preserved full binary compatibility with previous hardware and applications. Existing 32-bit applications will run absolutely unchanged and without performance degradation on AIX 5L. Furthermore, AIX 5L will run on all of the existing 32-bit RS/6000 systems that are still supported. This is important to many users because they prefer, for ease of management, to have all of the systems in their enterprise at the same

operating system level. With AIX 5L, 64-bit and older 32-bit systems can have the same operating system installed, and 64-bit systems can also run 32-bit and 64-bit applications at the same time.

Since AIX 5L provides either a 32-bit kernel or a 64-bit kernel (selection made at installation), a word about compatibility using AIX 5L is in order. The 32-bit kernel provides compatibility with existing 32-bit AIX applications, while the 64-bit kernel includes 64-bit device drivers and APIs, enabling increased use of system resources and larger application workloads on 64-bit hardware. Obviously, since most pSeries systems have 64-bit architecture, it is advantageous to install the 64-bit kernel in most situations.

From a compatibility standpoint, applications written to earlier AIX 32-bit versions are binary compatible and will run on AIX 5L (both 32-bit and 64-bit kernels), except in some unusual circumstances, such as the programmer having used undocumented software interfaces. Note that 64-bit applications produced using AIX Version 4 will not execute directly on AIX 5L. Those applications must be recompiled before they will execute on AIX 5L.

The next question you may have about your 32-bit applications is, "Should I recompile them to 64 bits to gain further performance advantage?" The answer is simple: Unless your application is under address space constraint, you will not improve performance by a recompile. In fact, all you will really end up doing is increasing the application footprint because your pointers take up twice as much space in 64-bit applications.

One last point about compatibility between 32- and 64-bit systems is this: Application developers can develop 64-bit applications even if they are using AIX 5L on a 32-bit system. The ability to build 64-bit objects does not depend on having 64-bit hardware. Of course, running these objects does require the full 64-bit environment.

AIX 5L has been designed to industry standards. The most predominant standard at this time is the Single UNIX Specification, Version 2 *(http://www.rdg.opengroup.org/public/tech/unix/version2/)*. This specification addresses such issues as the API, large file support (AIX 5L v5.2 supports files as large as 16 TB), and more. Compliance with this standard has been verified to the satisfaction of the Open Group, and AIX 5L has received the UNIX 98 brand. AIX 5L has also adopted the 64-bit data standard called LP64, as is shown in Figure 3.13. This defines for developers how many bits should be allocated for the various data types.

Data Type	LP64	ILP32
char	8 bits	8 bits
short	16 bits	16 bits
int	32 bits	32 bits
long	64 bits	32 bits
pointer	64 bits	32 bits

Figure 3.13. AIX uses the industry 64-bit data standard LP64. This chart shows the number of bits used by the different data types in the LP64 standard as well as the 32-bit ILP32 standard.

AIX 5L Summary

AIX 5L, together with the Bonus Pack, Expansion Pack, and Web Download Pack, provides an open, high-performance UNIX operating system with a rich set of networking and security features, systems management and performance analysis tools, and application development facilities. AIX 5L supports the most recent pSeries hardware technology as well as the latest open-system standards and technologies. As has been the case since AIX was introduced in 1986, AIX 5L will continue to be enhanced as hardware, software, and network technologies evolve.

The Linux Operating System

It's easy to see that the Linux operating system, once a phenomenon of academic or investigative interest, has moved fully to the stage of commercial implementation. Just walk through your local computer store and notice all the Linux distributions and Linux applications that are available. Or browse some computer software sites on the Internet, and it will be obvious that Linux is here to stay. Most of the Linux products in the stores are designed for Intel-based systems for small business and home use. But there are several distributors of Linux and suppliers of Linux services that provide the products and services that businesses require in order to embrace Linux as one of their operating system options—or perhaps as their only operating system.

IBM in general, and pSeries in particular, have embraced Linux and open source software as key components in taking e-business to the next level. Along with other open standards, such as HTTP, XML, and TCP, Linux plays a pivotal role in bringing interoperability to disparate server platforms and providing customers with an open, integrated e-business structure. Linux has evolved into an industry standard, and IBM continues to work with the Linux and open source communities to support open source software and related standards across its servers. Linux is UNIX related in the sense that it is based on the open-system design philosophy used in the original UNIX development, but Linux is not a UNIX operating system. Whereas UNIX has split into several competing UNIX operating system, such as IBM's AIX 5L and Sun Microsystem's Solaris, Linux is a distinct single-source operating system. Because of its strict adherence to open-system standards, Linux does not have all the extensions that have been developed for the various UNIX implementations, since those enhancements may depend on unique capabilities of one particular manufacturer's systems. But since Linux is an outgrowth of the UNIX heritage, it is a natural fit for companies that use UNIX servers to be interested in deploying Linux as well. And that's where the pSeries comes into the Linux picture.

IBM pSeries management believes that Linux will help drive the long-term growth of the Internet by providing an open application platform that can harness leading-edge technologies and simplify customer choice. The common application platform will help ensure software interoperability across heterogeneous servers. IBM is an active participant in the Open Source community, not only embracing its software, but also contributing significant skills, technology, and Open Source software where appropriate.

Background

Linux, named after its original developer, a Finnish computer science student named Linus Torvalds, is an Open Source technology. The Open Source community of independent developers provides source code that is available free for anyone to use. As stated in the Introduction, this does not mean "free" in terms of price (a company may charge for distributing the software). But anyone is free to copy the source code and give it away to someone else, free to change the software by having full access to the source code, and free to distribute an improved version of the software.

IBM has made and continues to make extensive commitments to support the Linux open computing environment by contributing software (mainly device driver code) to the Open Source community and by participating in activities such as the Linux Standard Base, the Free Standards Group, the Open Source Development Lab, and the Linux for the PowerPC Architecture project. IBM pSeries works with several Linux distributor partners—SuSE, Red Hat, and Turbolinux—to develop and test Linux distributions to run on the pSeries server family.

Linux on pSeries

All of IBM's server platforms are Linux friendly, including zSeries, iSeries, pSeries, and xSeries servers. xSeries servers are certified under the IBM ServerProven Program for Caldera, Red Hat, SuSE and Turbolinux distributions. In addition, selected models of IBM IntelliStations7 and ThinkPads7 are Linux enabled. For mainframe systems, Linux for zSeries and Linux for S/390 are available from distribution partners. IBM has enhanced the openness of the zSeries and iSeries systems by providing Linux interoperability that can extend file serving, data, printing, and other services to Linux clients throughout an enterprise.

The Linux affinity APIs in AIX 5L that support the recompilation of applications written for Linux for execution on pSeries products running AIX 5L have already been described. While this approach is useful in some circumstances, companies whose businesses depend on running Linux applications are likely to want to run Linux "natively," that is, directly on the pSeries hardware. There are several distributions available which run on pSeries. Recently, a Linux client for the Web-based System Manager was added in AIX 5L v5.2.

Generally speaking, Linux performance does not scale as well as AIX 5L for SMP servers with more than eight processors. However, as development evolves in the open-system community, this situation will change. If their requirement is to run Linux only, pSeries users should consider entry and midrange systems such as p615 (1- and 2-way), p630 (1- to 4-way), and p650 (2- to 8-way). If they want to run Linux in one or more LPARs, then they may choose from the complete range of LPAR-capable pSeries servers (p630, p650, p655, p670, p690).

One other consideration for users who wish to run Linux on pSeries is that I/O device and adapter support is more limited than when running AIX 5L. The reason for this situation is that device driver code for supported devices has to be ported for Linux and made available to the

Open Source community. Initial priorities for device drivers have been for SCSI, Ethernet, Fibre Channel, and 2D graphics adapters. Currently, Linux on pSeries can take advantage of Ultra-SCSI Differential and Ultra3-SCSI adapters (not yet RAID), up to Gigabit Ethernet adapters (UTP and fibre), the 2 Gbps Fibre Channel adapter, and the GXT135P graphics adapter (including digital support). A selected set of media devices and disk storage subsystems including IBM 2104 Expandable Storage Plus, IBM FAStT systems, and IBM Enterprise Storage Server (ESS) are also supported.

Linux Distributions for pSeries

SuSE was the first to offer a Linux distribution for pSeries and RS/6000 systems. After offering two PowerPC versions using the 32-bit kernel, SuSE introduced the 64-bit SMP-capable SuSE Linux Enterprise Server 8 (SLES 8) for iSeries and pSeries in December 2002. SLES 8 is now supported on all available pSeries and RS/6000 servers except the 32-bit RS/6000 Models B50 and 150. It is also supported in LPARs on the pSeries systems that provide that capability, namely p630, p650, p655, p670, and p690.

Red Hat introduced a Linux distribution for pSeries and RS/6000 systems with Red Hat 7.1 in early 2002, using the 32-bit Linux kernel. An update in December 2002 uses the 64-bit kernel, but still supports only 32-bit applications. Red Hat Linux 7.1 for pSeries (64-bit) runs on a few RS64 IVB and POWER3-based servers—pSeries 620, pSeries 640, and pSeries 660. Since these models are near the end of their life, this 32-bit distribution is of limited interest. IBM has recently entered an agreement with Red Hat with the goal of making Red Hat Enterprise Linux Advanced Server with a 64-bit kernel available to support both 32-bit and 64-bit applications for pSeries in late 2003.

Turbolinux recently introduced in Japan and is anticipated to announce in other Asia Pacific countries their Turbolinux Linux Enterprise Server 8 (TLES 8) product. Previously, the only Turbolinux distribution for pSeries was a 32-bit product for the p640 server. Turbolinux is a key distributor in the Asia Pacific market, and it is anticipated that TLES 8 will have broad support for pSeries systems in the near future.

IBM does not sell and support the Linux operating system, but several Linux distributors provide software, support, and services for pSeries

users who choose to install Linux. IBM Global Services will, for a fee, provide Linux services and support, also. As was mentioned in Chapter 1, IBM offers the pSeries 615, 630, and 650 servers in "Linux-ready" Express Configurations, which come without an AIX 5L license. Customers are responsible for purchasing and installing Linux. As a convenience for the customer, when one of these Linux-ready configurations is ordered, IBM will accept and pass through to SuSE an order for SLES 8. This pass-through order is available only for initial system purchases, and SuSE provides the software, license, and support.

IBM Applications Software for Linux on pSeries

A version of IBM DB2 Universal Database (UDB) for Linux on pSeries is now available. In addition to DB2 UDB, IBM has announced its intention to support many other software products on pSeries with Linux, including WebSphere Application Servers (powered by the Apache HTTP engine) and selected Tivoli systems management tools. Also, the IBM Journaled File System and IBM Developer Kit for Linux, Java 2 Technology Edition are certified on SLES 8 and Red Hat 7.1.

IBM is also working with its business partners and third-party software suppliers to port, recompile, and test their applications on pSeries platforms. Users should check with the appropriate software vendor regarding support of a specific application on pSeries servers.

Linux on pSeries Summary

AIX 5L, IBM's high-performance UNIX operating system, will continue to be the key operating system around which pSeries servers are developed and with which all pSeries innovations—processor and memory scalability, dynamic LPAR, dynamic processor and memory CUoD, RAS functions, and support for a wide range of I/O devices—are fully utilized. But IBM provides extensive support for users who need to incorporate the Linux applications into their pSeries computing environments, whether that means running Linux as the sole operating system or running Linux for selected applications. With respect to operating systems for pSeries servers, the key word is "choice."