ETA SYSTEMS

ETA^{10} SYSTEM OVERVIEW

PUB-1006
Revision A
February 28, 1986

Not all features described in this document are available in the first releases of the ETA10 system. Please consult one of the ETA Systems sales representatives for details. This product is intended for use only as described in this document. ETA Systems, Incorporated is not responsible for the proper functioning of undescribed features.

The items listed below are referenced in this document and are names, products, or trademarks associated with the following companies:

Name	Company
AT&T	American Telephone and Telegraph Corp.
B.E.S.T	ETA Systems, Inc.
CDC	Control Data Corporation
CYBER 205	Control Data Corporation
DEC	Digital Equipment Corporation
ETA	ETA Systems, Inc.
IBM	International Business Machines Corp.
LCN	Control Data Corporation
Open Interconnection Network	ETA Systems, Inc.
Multi Host Network	ETA Systems, Inc.
Multibus	Intel, Inc.
MVS	International Business Machines, Corp.
NOS/BE	Control Data Corporation
NOS V2	Control Data Corporation
PC/AT bus	International Business Machines, Corp.
Q-bus	Digital Equipment Corporation
Remote System Support Center	ETA Systems, Inc.
RHF	Control Data Corporation
Unibus	Digital Equipment Corporation
UNIX	American Telephone and Telegraph Corp.
VAX	Digital Equipment Corporation
VMS	Digital Equipment Corporation
VSOS	Control Data Corporation

Prepared by: ETA Systems, Incorporated
Technical Communication Dept.
1450 Energy Park Drive
St. Paul, MN 55108

Revision Record

New features, as well as changes, deletions, and additions to information in this document are indicated by bars in the outside margins.

Revision letters I, O, Q, S, X, and Z are not used.

REV./DATE	DESCRIPTION
Rev. A (86-02-28)	New document issued at revision A.

Pages changed in this revision:	
New issue.	

Table of Contents

LIST OF FIGURES

About This Document . . .

Introduction

This document is a general introduction to the ETA10 system. It presents more detailed information than you would find in a brochure, but less than would be included in a reference manual. The *ETA10 System Overview* is written from the user's perspective, rather than the system's perspective.

Intended Audience

Anyone interested in learning about what the ETA10 system is and what it does should read the *ETA10 System Overview*. The document provides basic information for many types of system interaction.

How This Document is Arranged

There are eight sections in this document:

- **Introduction** – This section discusses in a broad sense how the ETA10 system appears to users . It describes the system architecture and features at a general level.
- **Using the ETA10 System** – This section describes the ETA10 software from the user's viewpoint. It covers the different programming environments, and, for each programming environment, the available programming languages, debugging aids, and multitasking aids. The operator and system administrator environments are described, as well.
- **Networks and the ETA10 System** – This section explains the standard ETA10 network support and how the ETA10 integrates with existing networks and systems.
- **Internal Software Characteristics** – This section provides further details of the basic software features of the ETA10. It describes the logical structure of the software system and the internal system functions.

- **Hardware Components** – This section explains the hardware of the ETA10. It talks about the functional hardware elements of the machine, the individual performance characteristics of these elements, and peripherals.
- **Documentation for the ETA10 System** – This section talks about the documentation that accompanies the ETA10 supercomputer.
- **Customer Support and Training** – This section describes the support and training available for ETA10 customers.
- **For More Information** – This section describes some of the other documents about the ETA10 that are being developed.

How to Use This Document

This is an introductory document, meant to give the reader a general understanding of the ETA10 system. Some readers will want to read this document in its entirety, while others will want to skim it or read only a few sections of particular interest to them. This document may be read straight through or selectively.

Section 1

Introduction

Introduction

The ETA10 supercomputer is ETA Systems, Incorporated's initial product offering in the supercomputer marketplace. The ETA10 will have both general purpose features and high-speed calculation facilities. ETA Systems will add more features to the ETA10 as it is enhanced to meet growing customer needs.

The ETA10 system is composed of four well-integrated, mutually supportive elements: (1) software, (2) hardware, (3) documentation, and (4) support. These elements are illustrated in figure 1-1.

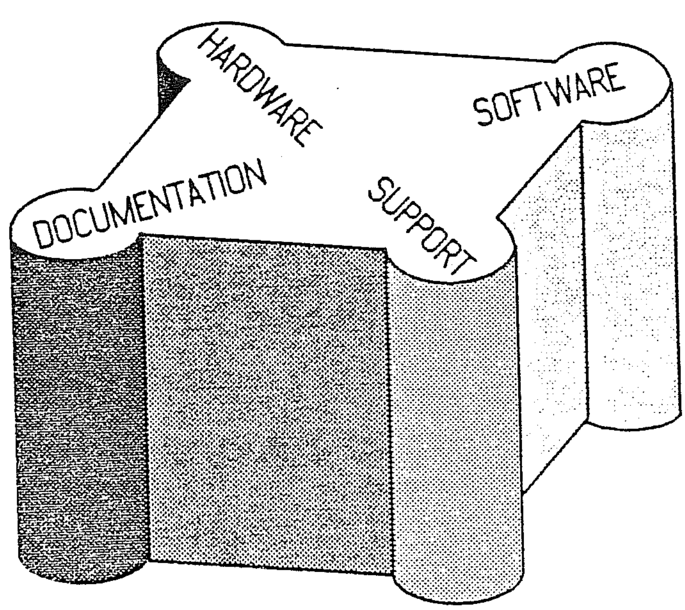


Figure 1-1. Interrelationship between ETA10 system elements.

User Views

An effective computer system meets its users' needs. This subsection shows how the ETA10 meets many types of user needs by providing tailored interfaces for different groups of users. This tailoring is like viewing the system through a lens.

Most of the first ETA10 users will be programmers. ETA Systems provides a choice of two programming environments (one which provides the UNIX user interface and tools, and one which provides the CYBER 205 VSOS user interface) that programmers can use to produce applications for the ETA10 supercomputer. The applications programmer sees the ETA10 system as being the programming environment he or she chooses. This is shown in figure 1-2.

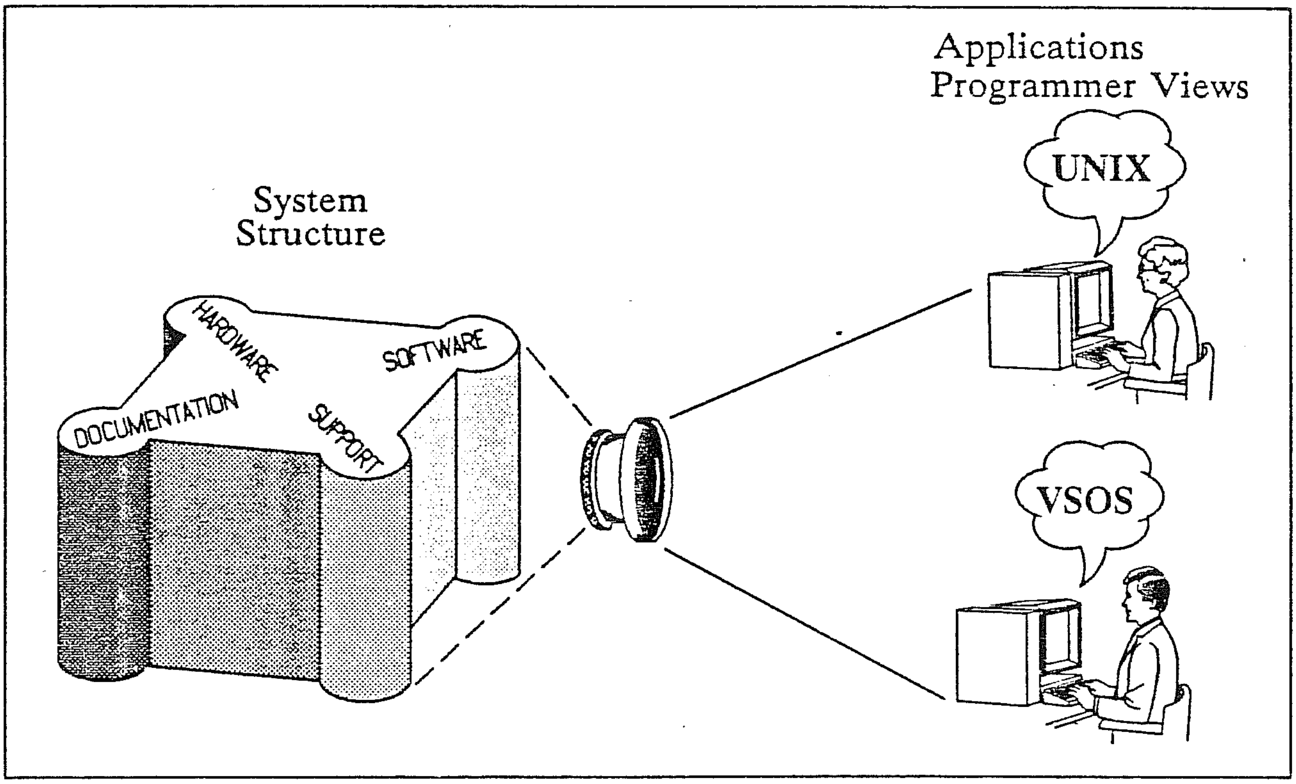


Figure 1-2. Applications programmers' views of the ETA10 system structure.

The next and largest groups of ETA10 users are applications users. They see the system through the lenses of their respective applications. Figure 1-3 illustrates this.

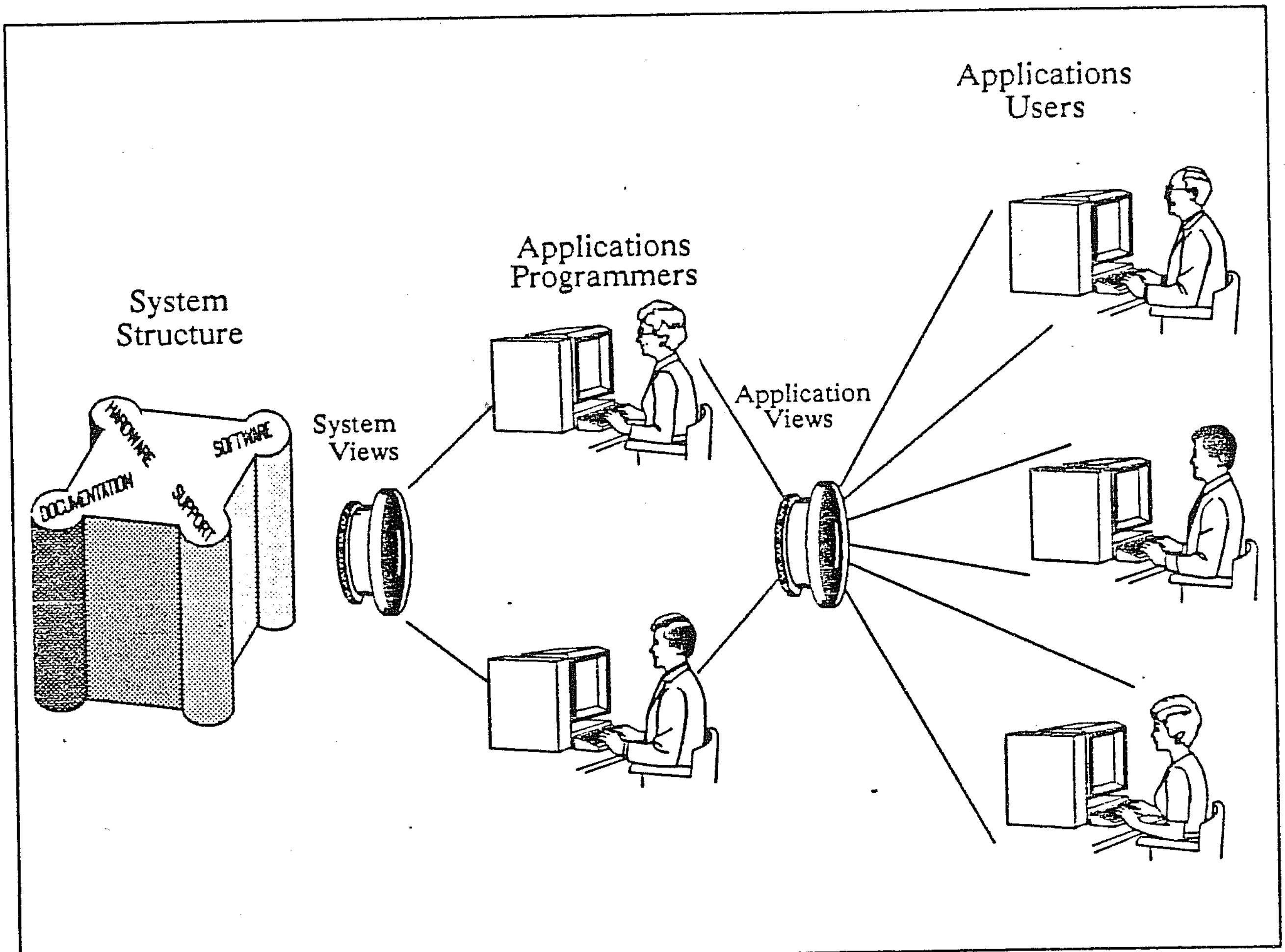


Figure 1-3. Tailored user views of the ETA10 system.

Some applications provide such complete user interfaces that the programming environments are not seen by the users. Other applications provide special services within a programming environment. For example, an application user may issue commands to the environment to stage his or her data to the ETA10 file system before running the application on the data.

By providing users with the appropriate lenses, users see the system in familiar and friendly ways. All users see the system's high peformance, reliability, and flexibility, in addition to the special features of their individual views.

System Features

This subsection introduces some of the major system characteristics of the ETA10.

ETA10 Hardware Architecture

As shown in figure 1-4, the ETA10 system has at least two, and as many as eight, Central Processing Units (CPUs). Each CPU is connected to the large Shared Memory (up to two billion bytes) and the eight-million byte Communication Buffer. Also communicating with these large common memories are from 2 to 18 Input/Output Units (IOUs) that provide connection to and control of all ETA10 peripherals and network interfaces. A Service Unit (SU) is connected to all parts of the ETA10, and provides an operator console and maintenance connections for the system.

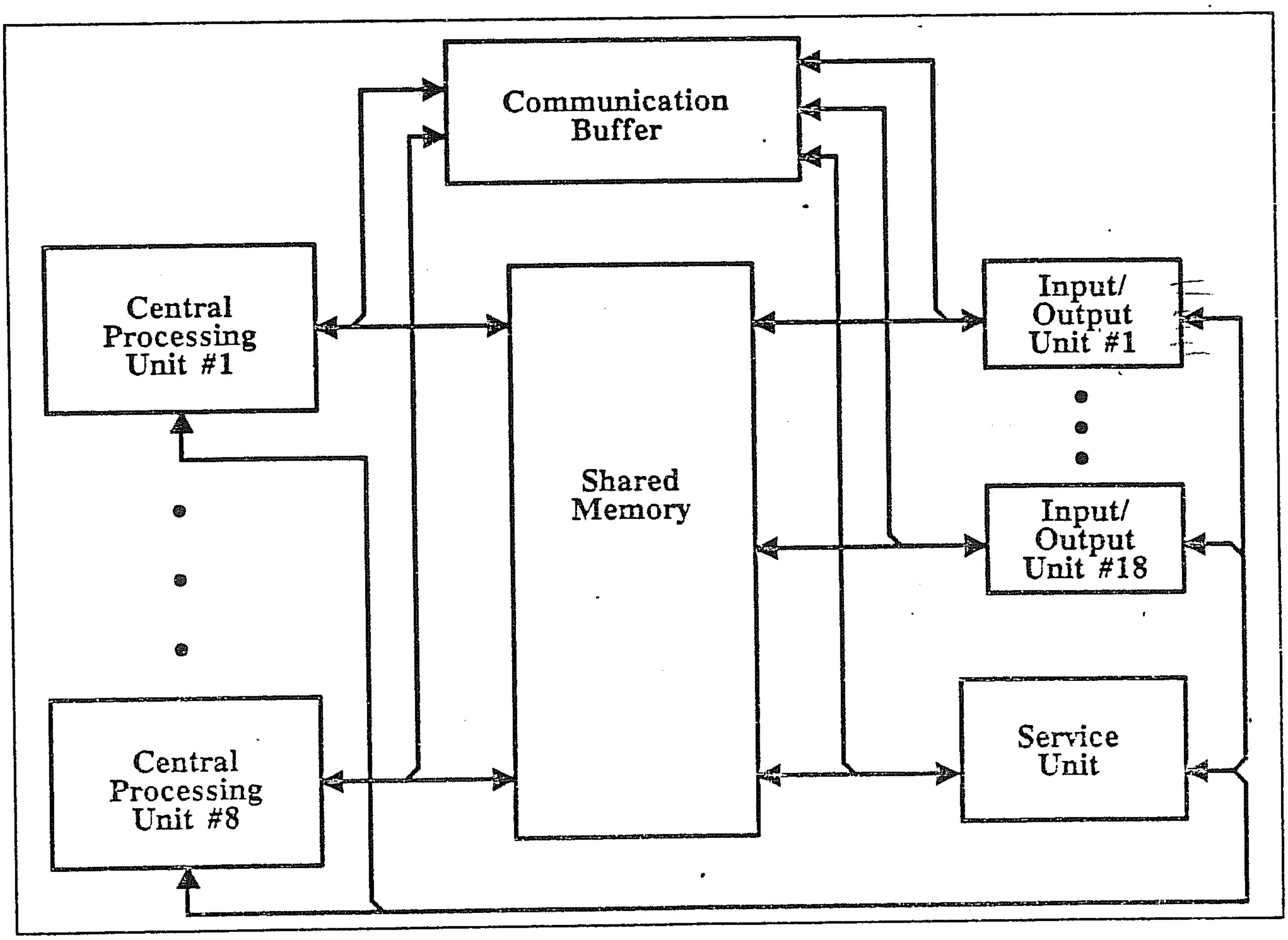


Figure 1-4. ETA10 supercomputer functional diagram.

Performance

The ETA10 system provides high performance for both computation and I/O, as described in the following subsections.

High Speed Computation

Each ETA10 CPU provides very fast scalar processing and vector processing. ETA FORTRAN provides state of the art optimization and automatic vectorization to system users. For applications with algorithms more readily expressed by vector notation, ETA Systems' ETA FORTRAN and Pascal compilers both will provide vector syntax.

The multiple processors of the ETA10 provide additional performance opportunities. Flexible system scheduling options allow the processors to be applied in parallel to provide very high performance for a single problem, or to be applied individually to sets of jobs to provide high system throughput. ETA Systems is developing techniques for the ETA FORTRAN compiler to perform automatic partitioning for multitasking of programs when a user selects this option.

Users who want to build explicit parallel operation into their programs also can use ETA Systems' multitasking library. Because not all programs naturally use the same model for parallel computation, this library contains several sets of calls, each of which provides a different way of expressing parallelism. One model is FORTRAN oriented, and avoids introducing shared data in a way that is contrary to FORTRAN's definition. Another model provides the counters and synchronization of the Industrial Real-Time FORTRAN subroutines. The programmer can select the model most natural and reliable for his or her application. ETA Systems will continue to add other models to this library.

High Performance Input/Output

The use of a large Shared Memory with a very high transfer rate between the ETA10 CPUs and disk subsystems provides the hardware needed to produce high overall I/O performance. Files being used are cached in the Shared Memory. The system selects buffer size, both in Shared Memory and the memory in each CPU, based on the user-chosen characteristics of each file. Users can provide file access hints to tailor the use of a file for higher performance by a specific program.

Installations can provide their users with very large disk files by configuring large logical devices from space on several physical disk devices. File size is limited only by the logical device size. Overflow of the data from one physical device to another is invisible to the file's creator. An installation also can configure the software to automatically distribute a file over more than one physical disk. When creating a file, the user only has to identify it as a "high performance file" in order for the software to distribute the data so it can be read or written in parallel using multiple disk channels.

Performance Analyzers

The ETA10 hardware provides individual processes with several counters that capture system information about a program's performance. Two examples are the scalar mode execution cycle counter and the vector mode execution cycle counter. The counters are accessible to a process by subroutine call.

In future releases, more sophisticated performance analyzers will be provided to give program tuning information about where programs spend most execution time and how they use memory. Tools also will be provided to help system administrators gather data on how effectively their system workloads are running.

Reliability

A reliable system is one that is available to its users when they need it, and one that correctly provides the functions they expect each time the functions are requested. The ETA10 system contains both hardware and software to make it reliable. The system has been designed to detect many errors on occurrence. Errors can be isolated to specific failing system components that can be removed from system service.

Because the ETA10 system can be configured with degradable hardware, such as multiple paths to a single disk, and multiple copies of the software, the back-up part of the system can be used, thereby providing a system that remains functionally available to users, even when part of it undergoes maintenance.

Fast isolation of an error also makes its recovery easier. The system performs recovery as transparently to users as possible. Work that can be restarted will be moved to system components that are operational.

Protection

Part of the system error isolation is provided by protection mechanisms within the system. The ETA10 hardware and software work together to create domains of protection that isolate different components of the system from each other. The user and system are located in separate domains, with controlled sharing of information between them.

The system provides other protection mechanisms, also. Processes running on the system are protected from each other, as are the files that they use. Installations control how much information flow is permitted between users on the system. For example, the standard UNIX programming environment is a relatively open system; users normally are able to send messages to each other and look up information about other users. While the UNIX programming environment can run in this standard way on the ETA10, there are options an installation can use to limit message traffic and information returned to a user inquiring about the system or about other users.

Reliable Documentation

A reliable system does what its documentation says it will. Both the on-line and printed ETA10 documentation provide accurate, easy to understand descriptions of the system, the hardware components, the software products, and their usage.

ETA Systems emphasizes on-line user assistance. In communicating with users, the on-line Help system uses the specific context in which the user is working when Help is requested.

Printed ETA10 documentation also takes into account the different levels of user expertise and interest. User guides give step-by-step information about the typical use of system features. Reference manuals provide in-depth descriptions of all feature options.

Program Error Detection

The ETA10 has features to help programmers make applications more reliable. Programmers have options to detect, trap, and attempt recovery of errors in their programs. This is true for arithmetical errors (for example, an attempt to divide by zero), as well as for other types of errors (for instance, executing illegal instructions).

Flexibility

The support of two different programming environments (the UNIX environment and the VSOS environment) as standard interfaces to the system is one example of ETA10 flexibility. Users can interact with the system in their customary ways.

Site personnel managing an ETA10 system see another aspect of its flexibility. An installation can define the classes of work it wishes to run on an ETA10, then dynamically select when the classes are to run. It will be possible to activate different workloads automatically at particular times of day.

An installation has a wide selection of choices for operating an ETA10 supercomputer. At one end of the spectrum, the ETA10 can be run almost automatically, with minimal operator involvement. At the other end of the spectrum, an installation can use specific operator interaction to control individual system events, such as causing a particular job to be run immediately.

Both the hardware and software systems are dynamically reconfigurable. Individual components can be added to and removed from the system without performing a general system restart.

Section 2

Using the ETA^{10} System

Introduction

As mentioned in the introduction, several kinds of users will likely interact with the ETA10 system. There are applications users, programmers, operators, and system administrators. While there are some people who are members of more than one of these groups, there are many people who prefer to concentrate their activities in one of the groups. It is useful to categorize members of different groups as having different areas of expertise, different expectations of system service, and different ways of interacting with the system. The ETA10 provides features and tools for each of these groups, using concepts and terminology familiar to them.

The majority of ETA10 users will be applications users. Some of the major types of applications that will be available on the ETA10 are structural analysis, fluid dynamics, nuclear applications, petroleum and geophysics, meteorology and climatology, chemistry, and electronics simulation and layout. These and other applications packages will be augmented by graphics, data base, and mathematical and statistical support. Because applications use varies greatly from one application to another, this document does not cover the use of specific applications.

ETA Systems sees increasing numbers of users who want to interact directly with their supercomputers from their workstations; interactive processing support is part of the basic ETA10 system design. The system also effectively supports the more traditional supercomputer approach of using it as a back-end, batch machine. Because users may work on the ETA10 either interactively or in batch mode, applications programmers can develop software that is delivered in the most appropriate manner to meet user needs. The ETA10 system also provides a variety of higher level languages and programming tools to support the development and conversion of applications for the ETA10.

Programming Environments

To help programmers create applications for the ETA10 supercomputer, the system provides two different programming environments: UNIX, a highly portable and widely used environment; and VSOS, the user interface of the CYBER 205. When programmers submit jobs or start interactive sessions, they choose one of the environments (or use the default the installation has assigned to them) for their particular jobs or sessions. Each of these environments provides the command language, tool sets, and the most important programming languages to which current users of these standard systems are accustomed. The components of a programming environment are shown in figure 2-1. Both these environments also have access to a set of ETA Systems products that are environment-independent.

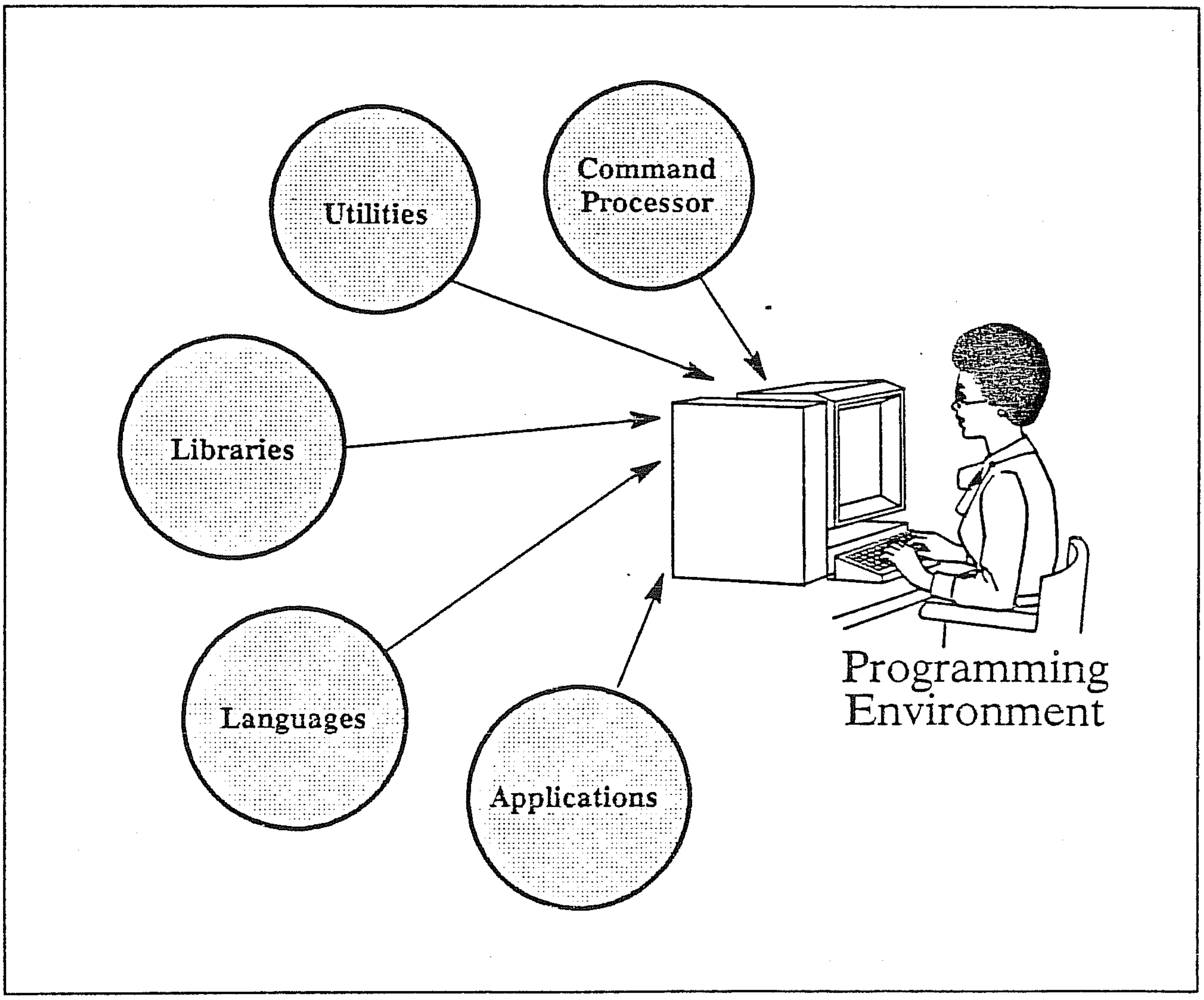


Figure 2–1. Components of a programming environment.

UNIX Programming Environment

ETA Systems has chosen to support a UNIX programming environment for several reasons: (1) it is compatible with many types of workstations, minicomputers, and microcomputers; (2) its user community is large and growing; many people know UNIX and write applications for a UNIX programming environment; and (3) it has a rich set of software development tools. The UNIX environment on the ETA10 is a full-featured software development environment, which emphasizes completeness and compatibility with the UNIX definition.

The UNIX programming environment on the ETA10 is based on AT&T System V, Release 2, with some Berkeley extensions. It has both the Bourne and C shells. Almost all the utilities and routines included in UNIX System V are supported, including the standard UNIX kernel calls. Additional networking facilities and accounting information also are available for users who wish to access them. As described later in this section, products common to all environments, including the numeric and multitasking libraries, also are available for use with the UNIX programming environment.

VSOS Programming Environment

The VSOS programming environment is provided on the ETA10 for compatibility with the CYBER 205. It is less full-featured than the UNIX environment, but the VSOS environment presents a straightforward system interface that is more explicitly tuned for batch performance than the standard UNIX interface.

The VSOS programming environment is based on the CYBER 205 VSOS Release 2.2. It has been enhanced in several ways. It provides richer interactive features, such as improved job control language with procedure file capabilities and multiple parallel sessions per user. The user interface also has been simplified by removing some CYBER 205 paging file limits and other file system limitations. Jobs have the ability to maintain file position from one task to another, simplifying the processing of input and output files. The owner of a file specifies the type of access allowed to the file, including multiple readers or multiple writers, if desired.

In addition to the common product languages described in the next subsection, the CYBER 205 FORTRAN200 compiler is provided in the VSOS programming environment for compatibility.

Also available on the ETA10 are many of the CYBER 205 System Interface Language (SIL) routines that provide an interface to operating system functions. This interface is conveniently callable by higher level languages. Most of the CYBER 205 SIL routines are provided; the exceptions are those that are dependent on CYBER 205 internal system table formats.

Products Usable from Either Environment

Several ETA Systems software products are designed to be used with either programming environment. For this reason, they are called common product languages or common product tools. These products include the standard programming languages (ETA FORTRAN, C, and, in the future, Pascal), multitasking capabilities, and some program development tools (debugging tools, for instance).

While these common environment products are functionally defined independently of the environments, they do use some of the environmental definition for context. For example, a user working in the UNIX environment can expect the standard debugger on the ETA10 to use the UNIX file search algorithms to locate requested files.

Programming Languages

In the first ETA10 system release, there are two common product programming languages: ETA FORTRAN and C. Pascal will be added later. ETA Systems sees FORTRAN and C as the early, most important languages for adapting applications for the ETA10 or for developing new applications.

FORTRAN

The ETA FORTRAN compiler supports the ANSI 78 standard language and anticipated array notation of the next ANSI standard. The compiler also supports some of the most popular language extensions from other major vendors, including BUFFER IN/BUFFER OUT, CYBER 205 FORTRAN vector syntax, IBM-compatible arithmetic type statements, and multiple assignment statements.

There are two ETA FORTRAN compilation modes on the ETA10: production and development. The production mode stresses execution speed and provides a variety of optimizations and vectorization. The development mode emphasizes compilation speed and extensive diagnostics. Users may select the compilation mode most appropriate for their needs.

Extensive scalar optimization is available to compiler users. The compiler provides common subexpression elimination, redundant load and store elimination, constant folding, removal of invariant code from loops, inner loop optimization, global register allocation, instruction scheduling, and other optimizations for scalar code.

A state-of-the-art automatic vectorizer also is included in the ETA FORTRAN compiler. The vectorizer not only finds the vector operations already in the code, it also applies numerically sound transformations to create additional ones. It recognizes vectorizable constructs in both DO and IF loops. When beneficial to execution speed, it will interchange loop indexes to create longer vectors or promote scalars to vectors. It also flags program statements where users can add directives to supply more information, allowing more vectorization than could safely be done automatically.

The compiler also provides limited automatic partitioning of programs in preparation for execution using multiple processors.

C

The C language, as defined in *The C Programming Language*[1], is fully implemented on the ETA10 system. The C compiler emphasizes compatibility with other portable C compilers, fast compilation speed, and diagnostics.

Pascal

A full implementation of Pascal, as defined in the *Pascal User Manual and Report*[2], will be available after the first release of the ETA10 system. The language will be extended to include a vector notation. The Pascal compiler will emphasize fast compilation speed and good diagnostics.

Other Programming Utilities and Tools

In addition to the tool sets within each programming environment, ETA Systems provides a variety of environment-independent utilities and tools. The system libraries, program development aids, and multitasking aids are detailed in the following subsections.

---

1 Brian W. Kernighan and Dennis M. Ritchie, *The C Programming Language* (Englewood Cliffs, NJ: Prentice-Hall, 1978).

2 Kathleen Jensen and Niklaus Wirth, Pascal User Manual and Report (New York: Springer-Verlag, 1985).

Libraries

A variety of standard support libraries is available on the ETA10. Some libraries are provided for use with a specific programming environment. These libraries (such as SIL for the VSOS environment, or kernel calls for the UNIX environment) provide functions that are tailored to program behavior in their particular environments.

Some other standard libraries provide functions that are not directly related to one environment or the other. These can be called by programs running in either environment. Some examples of these functions are mathematical and statistical routines, graphics routines, performance measurement, and multitasking functions.

In addition to using the standard system libraries, users can create and maintain their own object code libraries on the ETA10. Library building tools will be available in both environments to create libraries, to add and delete modules, and to provide the option to link them internally.

Program Development Aids

All common product compilers on the ETA10 provide extensive, compatible, symbolic information that can be used across languages by the ETA Systems debugger. The standard ETA10 debugger can be used either interactively or in batch mode.

The symbolic debugger assists users in debugging their ETA FORTRAN or C programs. The symbolic debugger also will work for Pascal programs once that language is available. Users can examine the actions of their programs by symbolic references to the original source code. The symbolic debugger allows users to debug in the context of the code they are working on, instead of showing them the hardware details (addresses, registers, etc.). Symbolic debug tools can be applied to programs with minimal effect on the programs' execution behavior.

In batch mode, the symbolic debugger provides ETA10 users with a postmortem dump facility. A postmortem dump is a snapshot of what the program was doing at the time it abnormally terminated. The system information is decoded back into the user's symbolic information about variables, files, line numbers, and so forth. Users can analyze this information to help determine what has gone wrong with their programs.

There also are system support functions that can assist users with multitasking debugging. For example, a user can specify that all processes in a session or job must be run in a single Central Processing Unit.

Multitasking Aids

In addition to the compiler option for automatic program partitioning described under the ETA FORTRAN compiler, ETA Systems provides a multitasking library to help users explicitly describe the parallelism in their applications. Users can access the multitasking library from either programming environment.

The multitasking library is an extendable set of basic routines that supports several models for multitasking. It acts as an extension to ETA10 common product languages to allow explicit use of the parallel architecture of the ETA10 system. The multitasking models define and manage the following three data types to implement the multitasking models:

1. A task is an independent, executable environment that is a subset of a user's program.
2. A counter is a shared semaphore with an integer value. It is used to provide mutual exclusion and to signal global events.
3. A shared array is used to share data between cooperating tasks of a user's program.

ETA^{10} Operator Environment

The ETA10 operator console is called the Service Unit (SU), and is the means of all operator communication with the ETA10. From an operator console, the operator can view system, component, and job status; control jobs; assign system resources; and reconfigure the system.

The operator interface is graphically oriented; menus and diagrams are used extensively, and technical terms are minimized. For basic operation, the operator needs no detailed knowledge of the system.

ETA10 System Administration Environment

The ETA10 system administration functions that require either special protection or the SU graphics capability must be performed at the SU, the same consoles that are used by the operator. However, most system administration may be done by system administrators working on interactive terminals connected by a network to the ETA10.

The system administrator defines how the system will be run and identifies users who are allowed access to the system. As is the case with the operator, the system administrator needs only minimal specific knowledge of the system in order to interact with it effectively. The system administrator's tools and documentation explain the necessary high-level system concepts in language familiar to system administrators. Menus and diagrams are used extensively in the system administrator interface.

Section 3

Networks and the ETA^{10} System

Introduction

This section is a general description of the options for incorporating and connecting an ETA10 system into networks of other computers and terminals. Two different network types are supported. The first, the Open Interconnection Network, provides interactive workstation and other network connections. The second, the Loosely Coupled Network, provides host-to-host high-speed communication.

Open Interconnection Network

The Open Interconnection Network supported by ETA Systems is based on the IEEE 802.3 standard for Ethernet. A single ETA10 Input/Output Unit (IOU) supports from 1 to 16 connections to the Ethernet cable(s). Each physical connection can support up to 128 logical connections to other devices on the network(s). Ethernet has a raw transfer rate of 10 million bits per second, and an effective applications data transfer rate of approximately 2 million bits per second.

The protocol supported over the Open Interconnection Network is the United States Department of Defense Standard TCP/IP protocol. This protocol facilitates the connection of a variety of devices to the Open Interconnection Network and the ETA10 system. The supported standard application protocols include File Transfer Protocol (FTP) and TELNET.

In addition to directly tapping an Ethernet Network, ETA Systems will offer a variety of devices that facilitate the connection of workstations, computers, terminals, and output devices to the network. These products will include:

- Protocol processing printed circuit boards and generic software for insertion into computers and workstations having an open architecture and a common bus interface, such as Multibus, Unibus, Q-bus, VME bus, or PC/AT bus.
- Terminal servers offering from 4 to 32 RS232, RS422, or V.35 ports for the connection of terminals, or for passthrough into other host mainframes supporting those terminals.
- Gateway servers supporting an X.25 interface, either to a public data network or to a host computer system supporting X.25 protocols.
- Bridges for connecting multiple Ethernet segments.
- Print stations and spooling software so print facilities are available directly from the ETA10.
- Network maintenance stations that provide control of network configuration, usage recording, directory services, and diagnostics.

The Open Interconnection Network also is designed to be compatible with other vendors' offerings that rely on Ethernet and TCP/IP. This design minimizes the constraints on customers in their selection of workstations to be used in conjunction with the ETA10. Some of the more popular workstations are certified by ETA Systems for operation with the ETA10.

Figure 3-1 depicts an example of an ETA10 system configuration with a variety of devices connected by the Open Interconnection Network. Nodes on the Open Interconnection Network can be reconfigured without disrupting other network activity. This includes reconfiguration around a malfunctioning node. Similarly, all maintenance and diagnostic activity can be accomplished without interrupting other nodes on the Open Interconnection Network.

As the International Standards Organization specifications for all seven layers of their Open System Interconnection model are implemented in the commercial marketplace, these standards also will be supported.

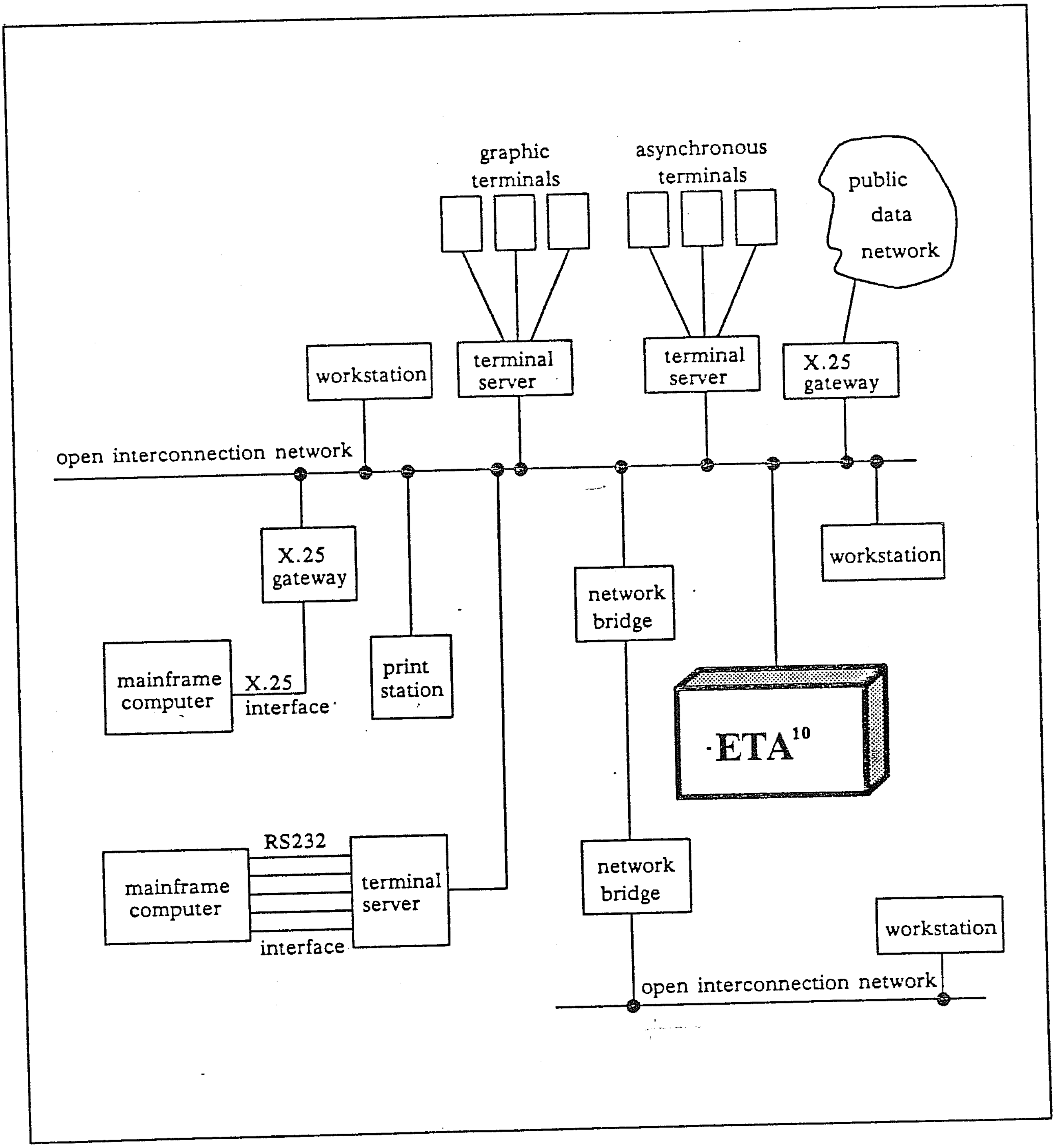


Figure 3–1. Example of an ETA10 system Open Interconnection Network configuration.

Loosely Coupled Network

The ETA10 system also can be connected to the Control Data Corporation Loosely Coupled Network (LCN). The LCN consists of various types of Network Access Devices (NADs) interconnected by coaxial cable using proprietary transmission mechanisms and protocols. NADs are available for connection to the following computer systems:

- CDC CYBER 170 and CYBER 180 series running NOS version 2.3
- CDC CYBER 205 running VSOS version 2.1.6

Connections to the these other computer systems will be verified later:

- IBM 303X and 43XX running MVS/SP version 1.3 with JES2/JES3
- DEC VAX models running VAX/VMS version 4.2

A NAD connects directly to a single channel on the CDC and IBM computer systems. The NAD model that interfaces to a VAX requires one slot in the VAX Unibus for access to memory. It can support up to four VAX machines.

A NAD can connect to as many as four different coaxial cables (trunks). Any single trunk can be up to 3000 feet in length or connect to as many as 28 NADs. The NADs provide contention-free transport services to the connected hosts, with a single NAD supporting up to 128 different virtual circuits. Figure 3–2 illustrates an example of an LCN configuration. LCN has a raw transfer rate of 50 million bits per second, and an effective transfer rate of up to approximately 6 million bits per second. (The effective transfer rate is dependent on the host system.)

The basic transport service provided by LCN is augmented with Remote Host Facility (RHF) applications running on each connected host computer. RHF provides for the copying of any file from any host to any other host. Copy modes can automatically perform character and record conversion to the native mode of the receiving host, or transfer the file as a transparent bit stream without conversion. RHF also supports entering a job into the input queue of a connected host, and returning output to the submitting host. Files also can be directed to particular output devices on any host.

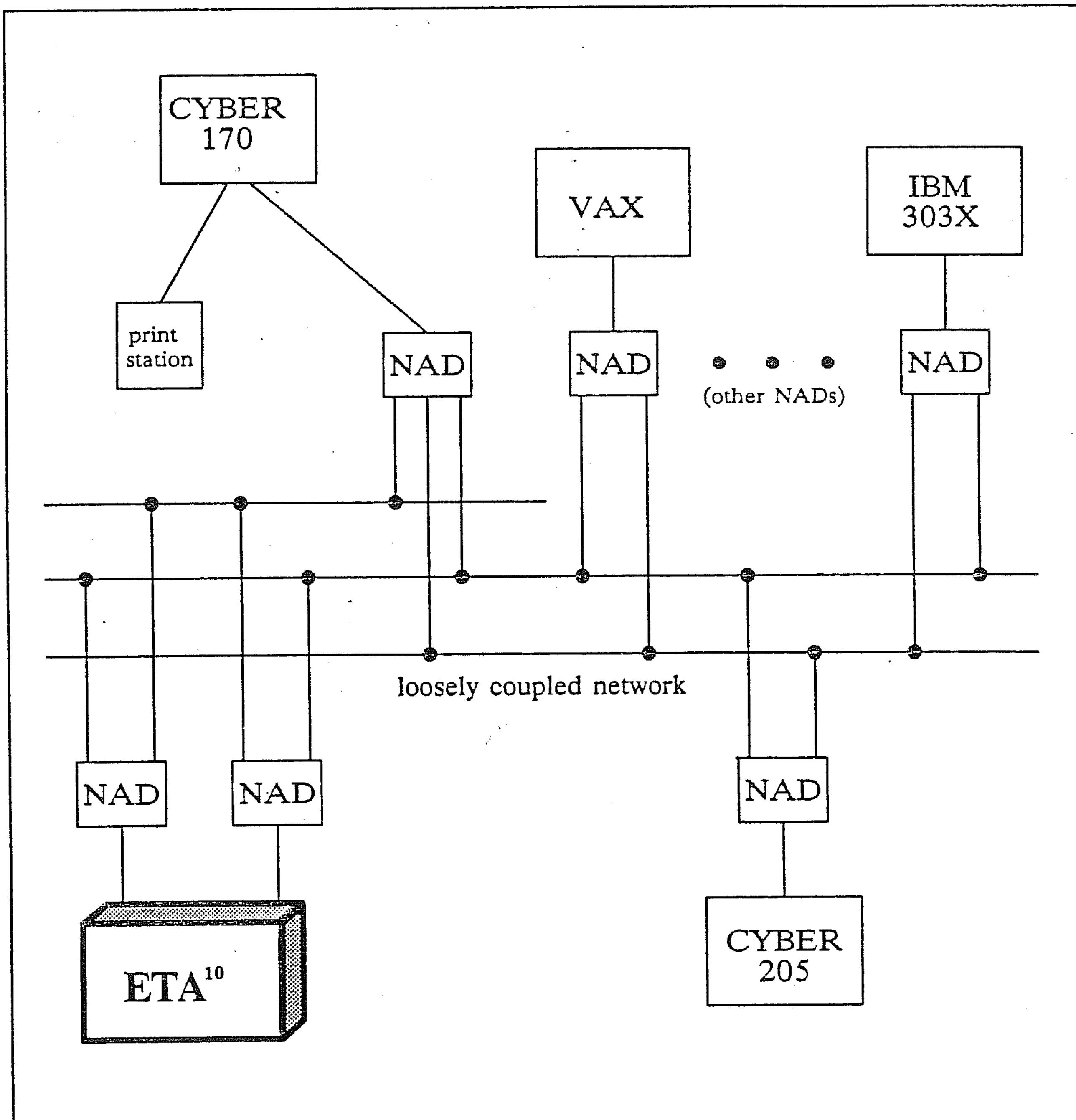


Figure 3-2. Example of a Loosely Coupled Network configuration.

Section 4

Internal Software Characteristics

Introduction

This section describes the logical structure of the ETA10 software system and the internal system functions that provide the features and protection needed by the user environments.

Software Structure Overview

The software system can be thought of as a set of concentric spheres. The outer sphere usually is applications. The greatest number of ETA10 users are expected to interact with the system at this level. The characteristics of the system are presented in the context of the job to be done. Figure 4–1 illustrates this.

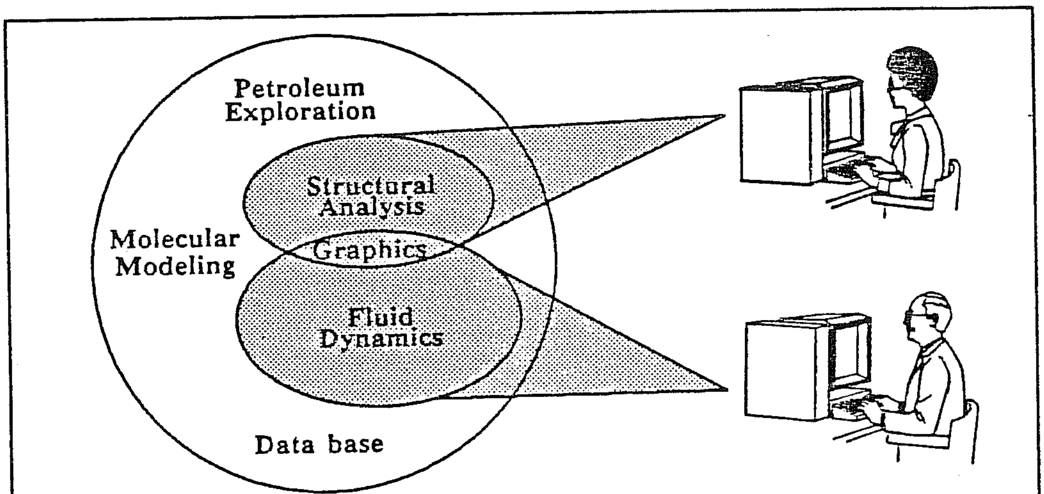


Figure 4–1. User views of the outer sphere of the ETA10 software.

Applications are directly supported by the next inner sphere, which contains the working environments (for programmers, operators, and system administrators), programming languages, and other application development tools. These software parts are designed to make applications easy to develop, use, and manage. The users of this middle sphere see the slice of it that they need to do their work. They also see interfaces to the underlying system kernel, which supports the applications. Some of these interfaces, such as UNIX kernel calls or SIL calls, are dependent on the working environment, while others are independent. All these interfaces are intended to provide consistent routine calls over time so changes to the most hardware-dependent ETA10 software need not force changes in already existing applications. Figure 4–2 highlights this working environment sphere. These system views were described in section 2, *Using the ETA10 System*.

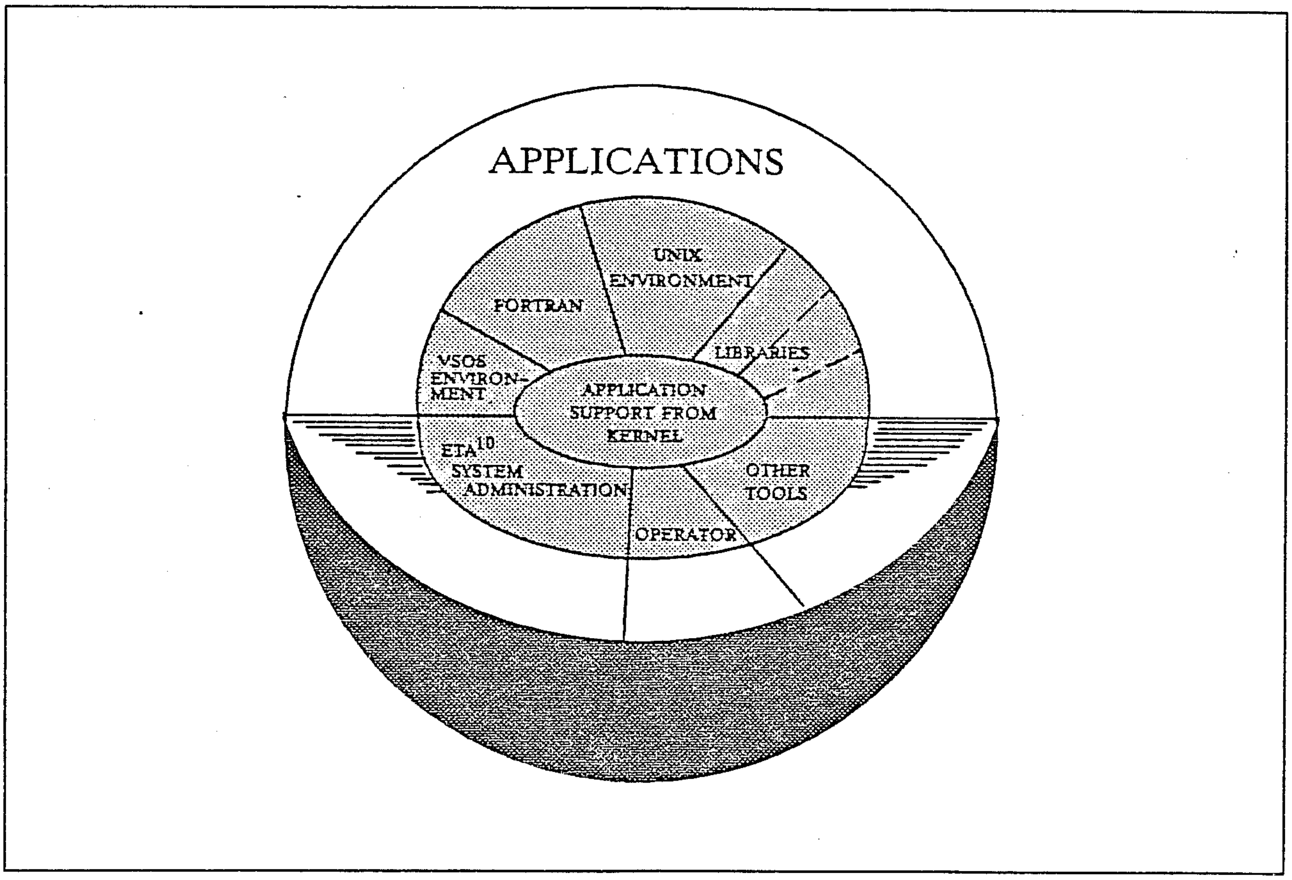


Figure 4–2. Working environment sphere (shaded) of ETA10 software configuration.

The innermost layer of the software system contains the kernel functions. It provides the specific control and protection functions required for the environments and management of the hardware functions of the ETA10 supercomputer. The full system structure is shown in figure 4–3.

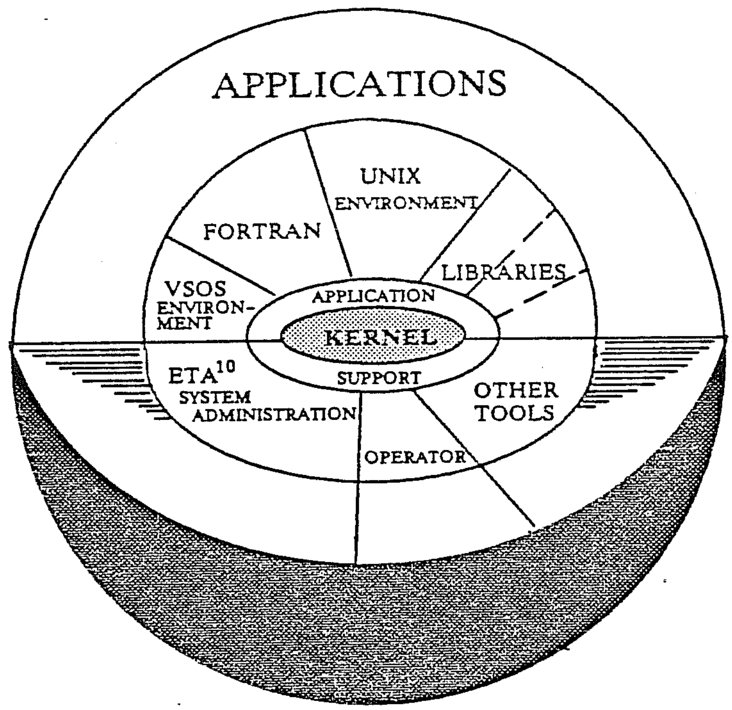


Figure 4–3. Full ETA10 software configuration (kernel shaded at sphere's center).

ETA10 Kernel Structure

The kernel level of the ETA10 software structure uses the basic hardware structure to implement basic system entities, such as processes and files, and to provide operations on them. The kernel is organized into features, each of which provides operations on its own set of elements that it defines and manages. Kernel features can be broadly categorized into three cooperating groups: the resource managers, the system overseers, and the system connectors.

Figure 4-4 illustrates the three groups of software and their relationships to each other.

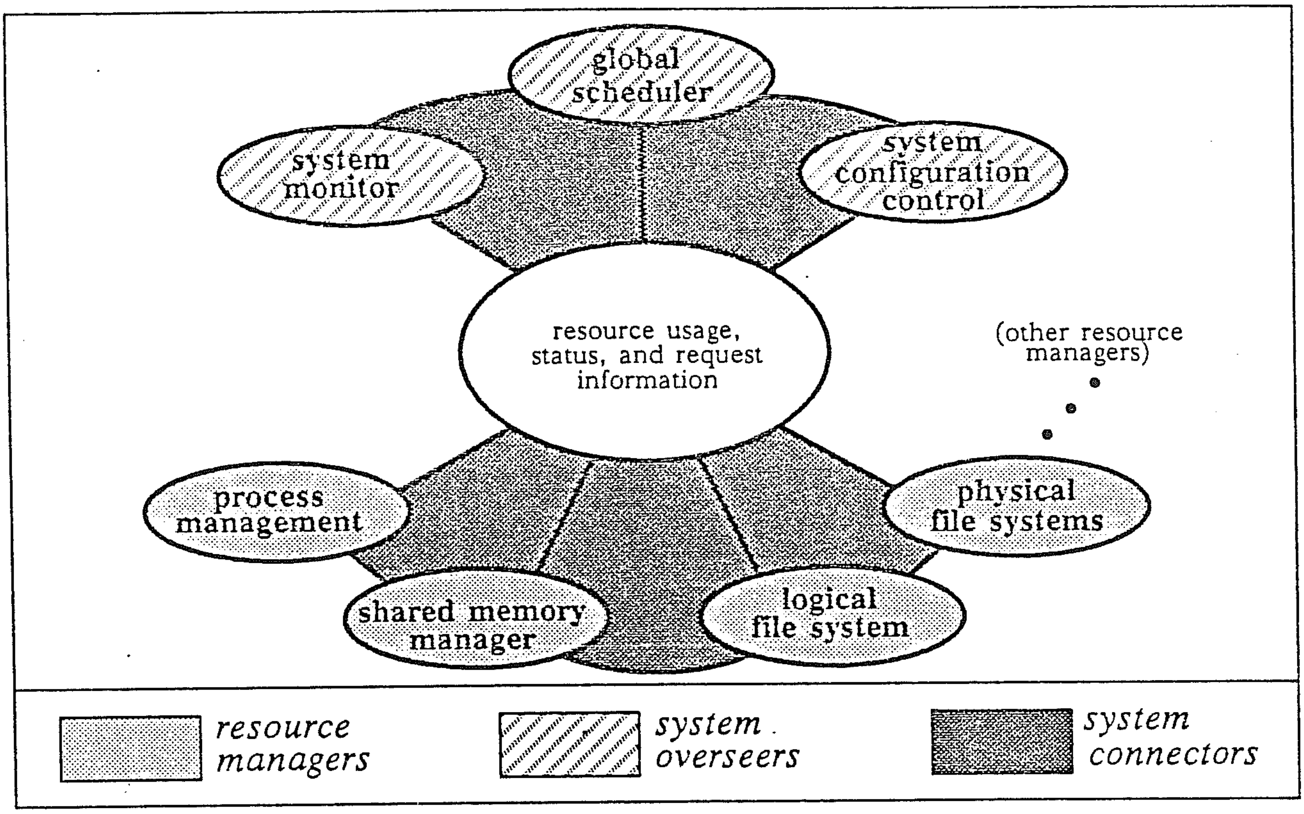


Figure 4-4. General categories of ETA10 kernel software features and their relationships to each other.

The resource managers control the individual architectural hardware pieces of the system (for example, the CPUs, Shared Memory, the Communication Buffer, the IOUs, and peripherals). The resource managers control processes and their scheduling, file creation and usage, use of the shared memories, allocation of disk space, and so forth.

The system overseers ensure that the system is running in an orderly fashion, operating efficiently, and getting work done according to the site's workload specification. Three of these activities are: (1) controlling system configuration, (2) monitoring system performance, operation, and reliability, and (3) global system resource scheduling.

The system connectors provide services that allow the parts of the system to work together in coordinated, protected ways. System connectors handle functions such as interprocess communication, remote procedure calls, and linkage between system modules.

Scheduling Options

System scheduling options are flexible on the ETA10. Both system administrators and individual users can tailor the system to meet their needs. The following subsections discuss the two types of scheduling options.

Administrative Scheduling Control

The system administrator uses special ETA10 management tools to define a set of desired site workloads. Then the workloads can be activated or deactivated by the system administrator or operator at the operator console. More than one workload can be run by the system at a time. For example, one installation may want to minimize the system response time for interactive users for part of the system's resources, while using the rest of the resources for large batch jobs. Figure 4–5 shows how two such workloads would coexist on a system.

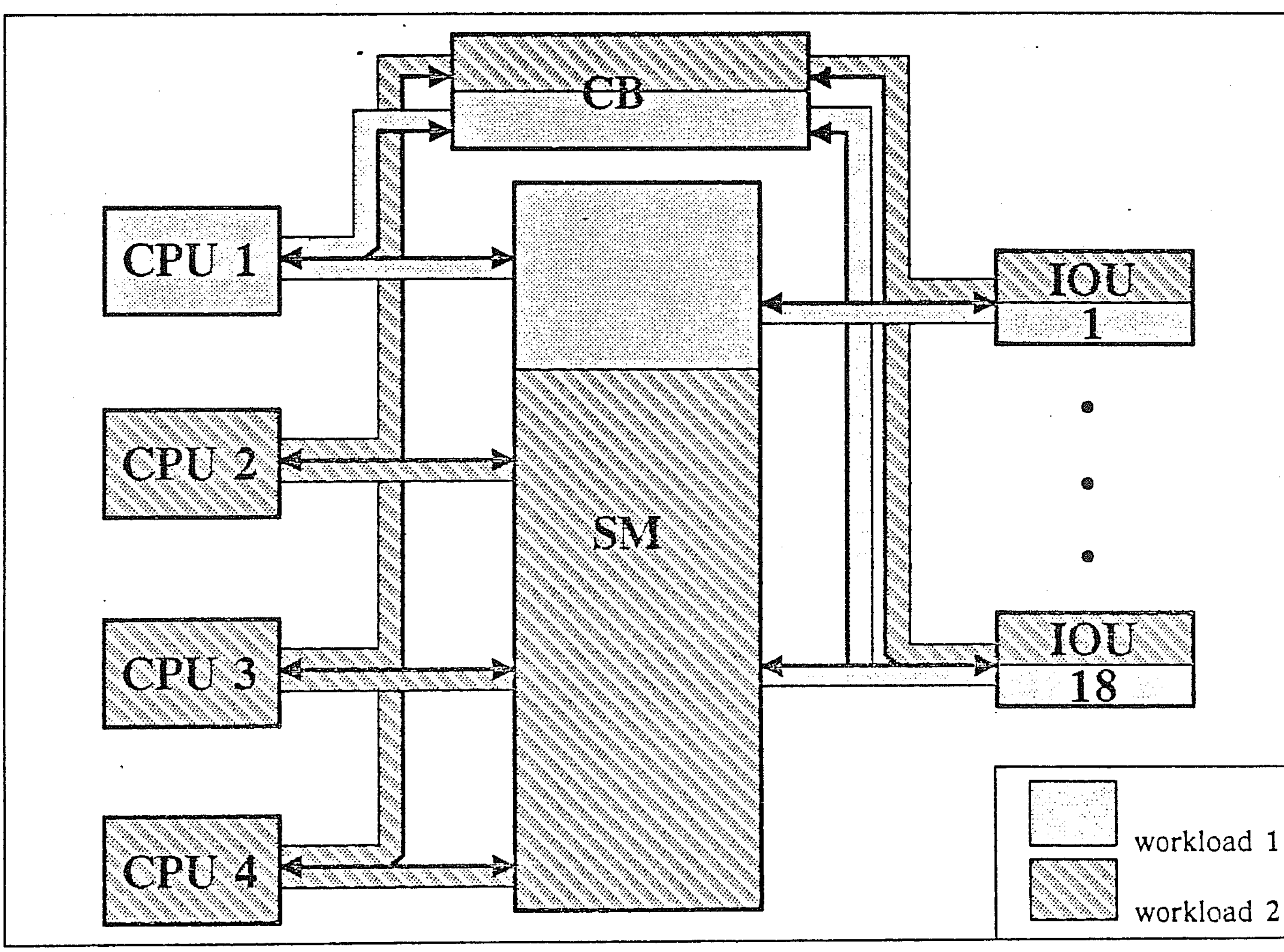


Figure 4–5. Two active workloads on a four-processor ETA10 system.

The ETA10 takes the system administrator's workload descriptions and turns them into sets of scheduling parameters. The CPUs have the largest number of scheduling parameters. Installations frequently want to select different time slices based on whether the workload is composed of small interactive processes or large batch jobs. Some installations may want the workload to affect the running page sizes, too. (See the memory management discussion later in this section.) The shared system resources, SM, CB, IOUs, and peripherals, are used more globally across the whole system load, though their use is influenced by ranking of the active workloads. When users submit batch jobs to the input queue, or when they attempt to log in, the system automatically follows site scheduling instructions when selecting which jobs to run and how to rank their resource use. The system monitors its own performance and resource usage, and balances the system workload according to the workload parameters.

User Scheduling Control

There are facilities in both programming environments for users to effect, within a workload, the suspension, destruction, and eligibility to run of their processes or jobs. The specific commands used depend on the user's selected programming environment. Users can enter work into the system input queue to be performed later. UNIX uses BATCH and AT commands, while VSOS uses the SUBMIT command. Users also can interact with executing jobs and processes as described in the next subsection.

Session/Job and Process Management

This subsection describes the two levels of control that users have over their active work on the system. One is a session or job level of control, and the other is a process level of control. A session or job is a set of serial or concurrent processes and system commands that do a set of work for the user. Accounting information is available for a session or job as a unit, as well as for its individual processes. The session or job level of control associates a set of processes with either an interactive session name or a batch job name. Users can monitor and control the set of processes as a single unit by using the session or job name; for example, if they want all processes in the job terminated at the same time, they can terminate the job name.

A process is the most basic unit of work and unit of resource allocation for the system. The process level of control presents each process as a separate activity to which the user relates directly. The user monitors and controls each process individually.

The job or session level of control is historically most often associated with the VSOS environment, and the process level of control usually is associated with the UNIX environment. However, either level of control is available, regardless of the environment.

While the levels of user interaction with work are usually different between the VSOS and UNIX environments, all programmers performing work on the system use a structure composed of both levels of system interaction. Historically, the term "job" often is applied to batch work, while the term "session" is applied to interactive work. Figures 4-6 and 4-7 show typical structures for interactive sessions and the processes associated with them in the UNIX and VSOS environments.

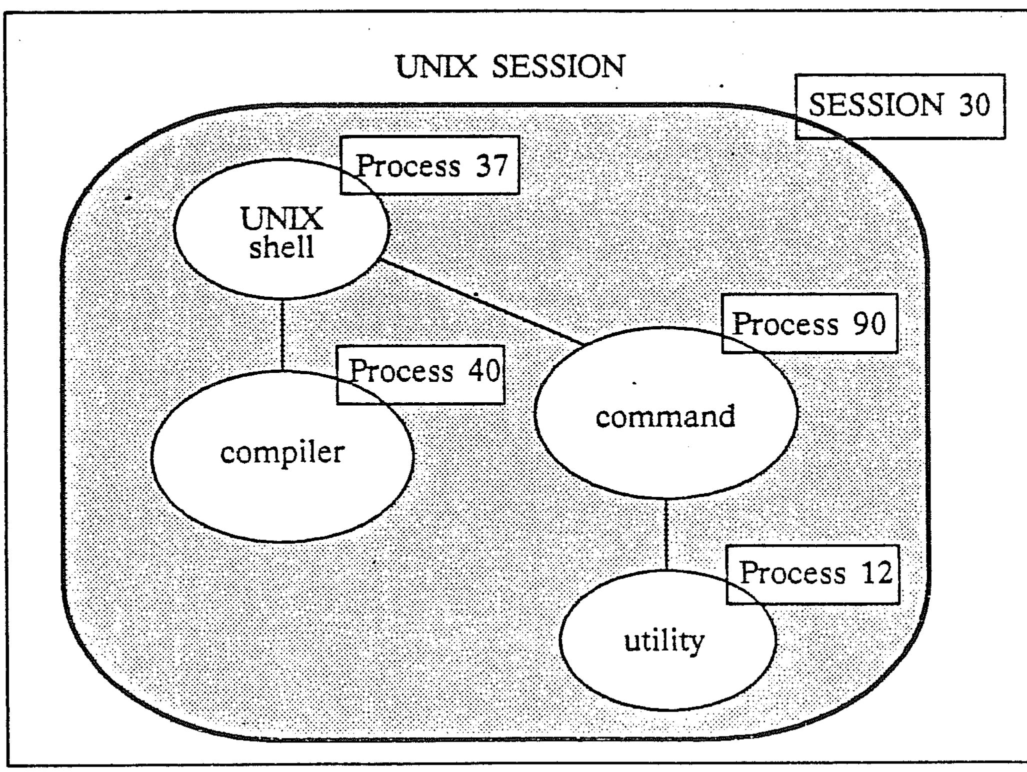


Figure 4-6. Typical UNIX interactive session snapshot.

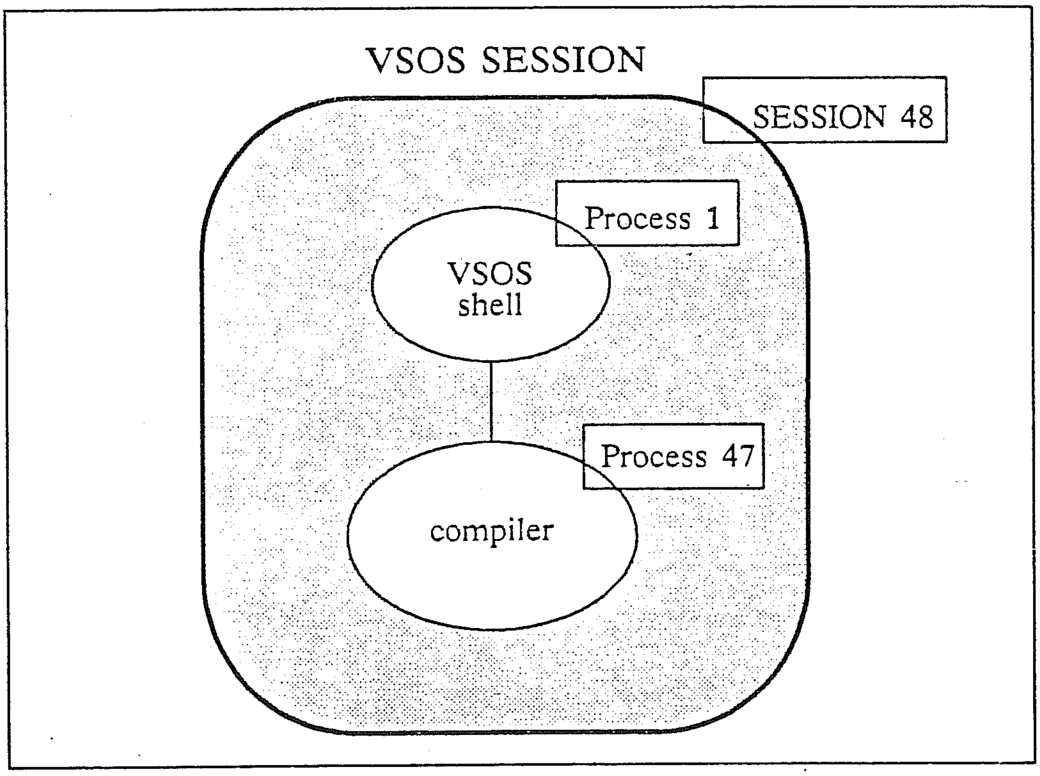


Figure 4–7. Typical VSOS job snapshot.

A user may take the following actions on a session/job or process:

- initiate or resume the session/job or process.
- disconnect from and reconnect to an interactive session.
- request the status. (Is the job in the input queue, in active or suspended execution, or in the output queue? Is the process running or suspended?)
- change some of the running parameters of the job or process.
- suspend or terminate the job or process.

Memory Management

The following two subsections explain how the system manages memory on the ETA10 and how users can use memory effectively in their programs.

Virtual Memory

Programs running on the ETA10 use a large virtual address space for their code and data. The address range is 0 to $2^{48}-1$ bits. All user programs and the great majority of the ETA10 system code run in this address space.

Hierarchical Memory Architecture

The underlying support for the large virtual address space is composed of four hierarchical memories. As shown in figure 4-8, these are a CPU's register file, its physical Central Processor Memory, Shared Memory, and disk storage.

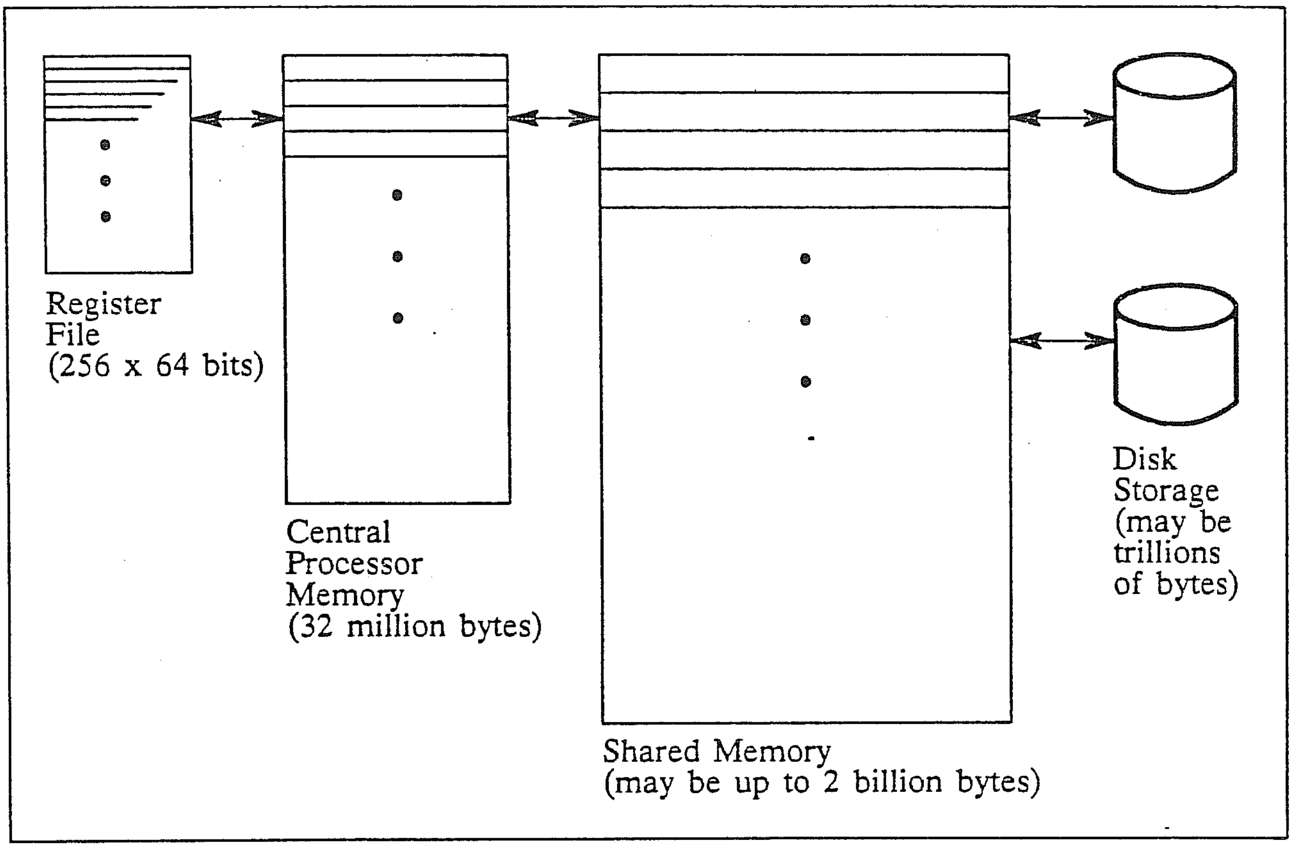


Figure 4-8. ETA10 memory hierarchy.

The large register file is managed by the compilers running on the ETA10 system. They generate the Central Processor Memory load and store instructions that manipulate the user's data using the register file.

Central Processor Memory has both hardware and software support for its virtual memory. Memory is organized into pages, and the virtual pages being actively referenced by a process at a point in time are mapped into physical pages in the memory. The maps are stored in a page table for the process. Each page being used has a key assigned to it and a set of access permissions (some combination of read, write, execute, or none). This information also is stored in the page table. For a process to reference a page in physical memory, the process must have the correct key and access permission, or the reference will cause an error. As described in section 5, *Hardware Components*, each CPU uses one small page size and one large page size at a time.

When a process attempts to reference code or data in a virtual page that is not resident in physical memory, a page fault occurs. The running process is interrupted, and, if the reference is valid, the system fetches the page from Shared Memory or disk, assigns it to a physical page of memory, and resumes execution of the process. The interruption is transparent to the running process except for the wall clock time needed to service the interrupt and obtain the page. The system uses a working set algorithm to attempt to keep the most likely used pages in Central Processor Memory.

Shared Memory contains buffers for Central Processor Memory paging files and data files that are being read and written by processes executing on the ETA10. Paging-operations between Shared Memory and Central Processor Memory are very fast because of the high bandwidth between these two memories.

The largest and slowest memory on the ETA10 system is on disk. When a page resident in Shared Memory is to be replaced with other data, that page is written to the process's paging file on the disk.

Communication Buffer

The CB is not a direct part of the virtual memory support system. Instead, it provides fast locking and synchronizing functions for the system. The multitasking library provides access to these functions.

User Interactions with Memory Management

Most user programs do not need to provide the ETA10 memory managers with specific memory use information. However, for the small percentage of users that have unusually high performance requirements, the ETA10 provides two types of memory requests: advice and demand.

Advice tells the system that in the next part of process execution, a particular range of addresses will be used, and perhaps that another range of addresses will not be used. If physical pages are available for the virtual addresses being advised into memory, the system will page them into physical memory. The process's execution then can continue without waiting for page faults later. If the needed physical memory is not immediately available, the process will resume execution after the advise request, then fault for the pages later when the explicit memory references are made. Advice can be given for memory ranges in Central Processor Memory, as well as for file buffer space in Shared Memory.

A demand is a request for a guarantee from the system that a specified amount of memory will be available for a process to use. If the full amount of memory is not available, the process will wait until its demand can be granted. A demand request to the system indicates memory use or data consumption that is so high that the process will not run effectively without the requested resources. Demands can be made for file buffers in SM.

File System and I/O

The physical file and I/O devices on the ETA10 are disks, networks, and tapes. The use of one device type or another is largely transparent to file system users writing in a common product language, except for the specific device characteristics. For example, tape files are used for sequential access, not random access.

Maximum file sizes can be set by the system administrator and by individual file owners. The system administrator can specify the maximum file size that a user is allowed to create, as well as the total maximum amount of disk space the user can have. Individual users can specify the maximum size, smaller than site limits, for each file they own. This feature helps users control runaway programs.

The allocation of space to files is dynamic. The system automatically extends a file while it is being written as long as there is space for it to grow on its logical device (described later in this subsection) and it is within the limits stated above. The system allows the user to preallocate file space, but it is rarely necessary for users to do this.

As described in the previous subsection, the path I/O takes across the hierarchical memory system is between CPM, SM, IOUs, and peripherals. The data transfer rate between peripherals and SM is high. The peripheral to IOU channels are fast. They are multiplexed into a faster IOU to SM path. However, the CPUs can use data faster than this. SM acts as a large, very high-speed peripheral for the CPUs. Files being used by executing processes are cached in SM buffers. There are two levels of I/O performed on the ETA10 system.

One occurs between Central Processor Memory and Shared Memory. The other occurs between Shared Memory and peripheral devices.

In CPM, the ETA10 supports both paged I/O and buffered I/O. To use paged I/O, the programmer maps a region of memory to a data file and references addresses within that memory region. If the data has not yet been brought into physical memory, the process uses page faults to make it available as described earlier in the Memory Management discussion. The programmer uses assignment statements for data structures in the mapped memory region instead of explicitly coding read or write statements.

Buffered I/O uses explicit read and write statements to cause the user's data to be moved to or from system managed buffers, which are automatically written when full, or read when empty. Unless paged I/O is explicitly coded, buffered I/O is performed automatically when read and write calls are issued from common product programming languages. For example, formatted or unformatted I/O in FORTRAN normally uses two buffers per file; it fills one while the other is being emptied by READ statements or empties one while the other is being filled by WRITE statements.

For buffered I/O, the customary file structures (records or byte strings) in each of the user environments are provided by the file structure of the ETA10. The ETA10 I/O operations have a set of record-oriented interfaces and a set of byte-oriented interfaces. The explicit record types supported are variable length, fixed length, record mark delimited, and unstructured. The byte-oriented operations access the file as a continuous byte stream.

In SM, buffering is used for all I/O. Buffer sizes are set as they are for CPM, by file characteristics plus user hints. The system sets its I/O buffer sizes by looking at the characteristics of the file. For example, if the data file is a high-performance file, the system attempts to provide bigger buffers. Users also can give the I/O system other file access hints, such as "read ahead" to tune the buffering.

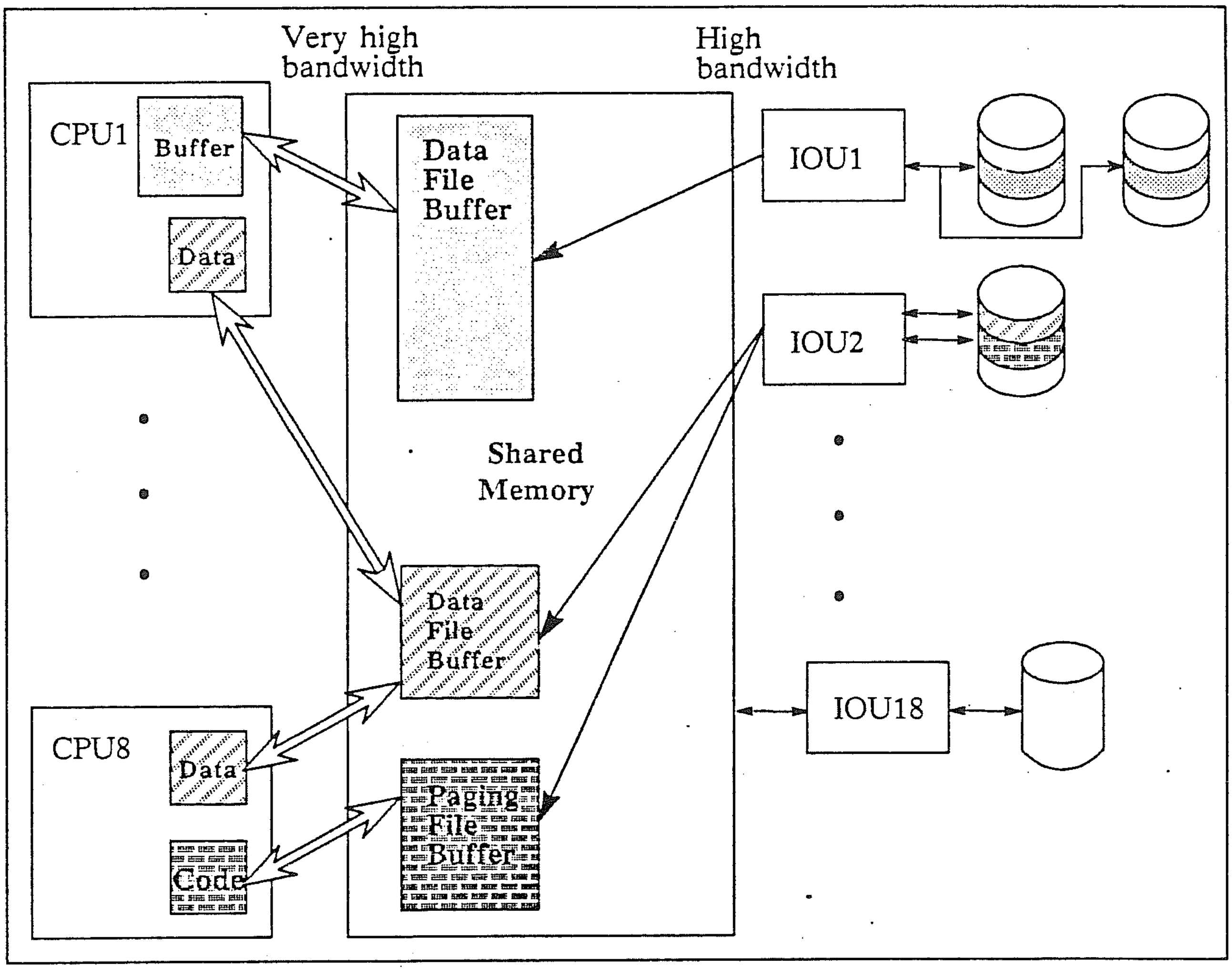


Figure 4–9. Buffering and paging of files on the ETA10.

Figure 4–9 shows several files in use. The file on IOU1 is a high performance data file. It is being accessed via buffered I/O with large CPM buffers and very large SM buffers. The first file on IOU2 is a typical data file. It is being accessed in two different CPUs via paged I/O. It has typical buffer sizes in SM. The other file on IOU2 is a process's paging file. Its CPM buffers are the virtual addresses that have physical memory currently assigned to them, and its SM buffers hold recently paged out process information.

The system file names and searching conventions are dependent on the environment being used. The underlying kernel software provides a hierarchy of directories with long directory names and file names.

Disk space configuration is available in a flexible way on the ETA10. System administrators define disk space configurations in terms of logical rather than physical devices. Logical devices can include multiple physical devices. A file is restricted to one logical device, but may grow very large as the site configures more physical disk space into the logical device where the file resides. The overflow from one physical device to another is managed automatically by the software and is not seen by the programmer. Figure 4-10 contrasts the physical distribution of information as seen by the system with the logical distribution of information as seen by the user.

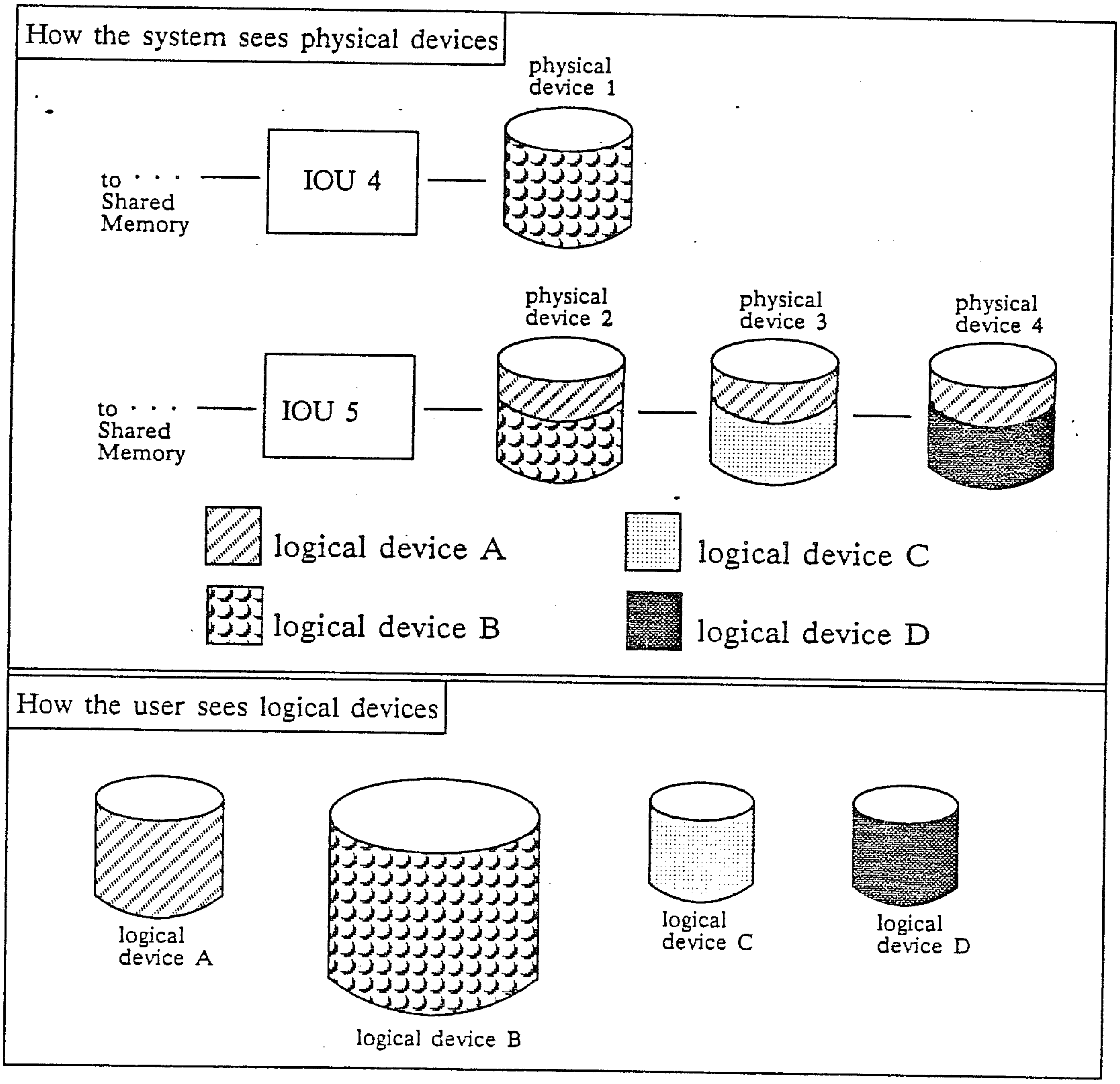


Figure 4-10. Physical distribution of information as seen by the system vs. the logical distribution of information as seen by the user.

The ETA10 connects to a variety of networks (described in section 3, *Networks and the ETA10 System*). Files can be connected to and disconnected from interactive users' terminals, and written or read much like other device type files.

The tape file system on the ETA10 supports unlabeled tapes and level 2 of the ANSI X3.27.1978 label standard plus creation date. EBCDIC-coded labels are supported in the same manner as ANSI standard labels. Translation to and from internal ASCII characters is done by the system. In its initial release, the ETA10 will support single-volume, single-file labeled tapes. After the first release, tape processing will be upgraded to support multi-volume, multi-file tapes. User processing of the following optional label types also will be supported: UVL-1, UHLa, EOV2-9, EOF2-9, and UTLa.

The system checks the first record of all tapes to determine whether or not a label exists. Labeled tapes are automatically recognized by the system when they are mounted, and assigned to the jobs requesting them.

Linkage Facilities

Linking object code and library routines can be done either as a separate step before execution or, in the future, dynamically during execution. Many user programs need only the standard system libraries to satisfy all of their calls. However, users also can instruct the system to form linkages to a list of libraries, and they can specify an order in which the libraries will be searched. When a program needs special versions of particular routines, a library name can be specified as a source of individual routine names.

The linking facilities, in conjunction with other related system support, provides options for users to reserve virtual address ranges or to assign them to specific data structures. It also allows users to group structures together to build particular memory reference patterns for the virtual memory. For example, it may be beneficial to execution time for a program using two arrays at the same time to put both of them into the same virtual memory page.

Libraries

A library is a collection of software routines with a directory. Usually the routines are related, such as commonly used mathematical functions. Libraries may be available for use by everyone at an installation, or they may be restricted to individual users. Libraries facilitate keeping compiled code in a usable form, so users compile only the new code they need, then add or replace it in an existing library. The system library is built so processes in a single CPU can share the same copy of the system code, leaving more space for user code and data.

ETA[10] Operator Software

The Service Unit (SU) is the ETA10 operator console and is the means for all operator communication with the ETA10 system. The SU may have from two to eight display stations, each of which can b an operator console.

From an operator console, the operator can:

- View system, component, and job status.
- Control jobs.
- Assign system resources.
- Reconfigure the system.
- Send and receive messages from users or processes.
- Mount tapes, etc.

Within a wide range, an installation may select the amount of interaction between the system and the operator. The system can ru with minimal interaction between the operator and the ETA10, or the operator can exercise detailed control over the system operating parameters.

The software that provides the functions used by the operator is distributed between the SU and the CPUs. Software functions that communicate among different processors do so by remote procedure calls.

ETA10 System Administration Software

Some ETA10 system administration software must be run on the SU, while other parts can be accessed from interactive terminals connected by a network to the ETA10.

The ETA10 system administrator can:

- Define user profiles, including their associated permissions, default user environments, and prolog (logon) and epilog (logoff) command sequences.
- Define and maintain the accounting structure for the system.
- Allocate system resources to accounts and projects.
- Select system parameters and algorithms used in charging for the use of system resources.
- Define site workloads.
- View system-related performance data.
- Select settings for site-specific options (for example, whether or not ANSI labels are required on all tapes).

The system administrator software is composed of a set of applications designed to use concepts and terms familiar to a person with administrative expertise, rather than programming expertise. The structure of the administrator environment is different from the programming environments in that there is no single command processor or "shell." However, the applications themselves present a uniform interface from one to another. Like the operator software, the system administrator functions are distributed between the SU and the CPUs.

System Management

The following subsections talk about the management of system resources on the ETA10. The first topic considered is the accounting structure of the ETA10 system. Next, the control of system resources is presented and explained. Finally, user access to the system is addressed.

Accounting

The accounting system on the ETA10 has two hierarchical levels of cost tracking and assignment: accounts and projects. Projects are allocated under accounts. Installations with less complex accounting needs can use the accounts alone, without projects. Users are assigned to accounts and projects. Each user has a default account/project pair, and also may have other pairs he or she can use. Units of charge are called "system billing units." These units are the sum of each resource used multiplied by an installation-selected factor for the resource.

Resource accounting is managed by a system administrator and multiple project administrators. There is a hierarchical relationship between these roles, with the system administrator being the overall controller. The structure of the accounting hierarchy is shown in figure 4–11.

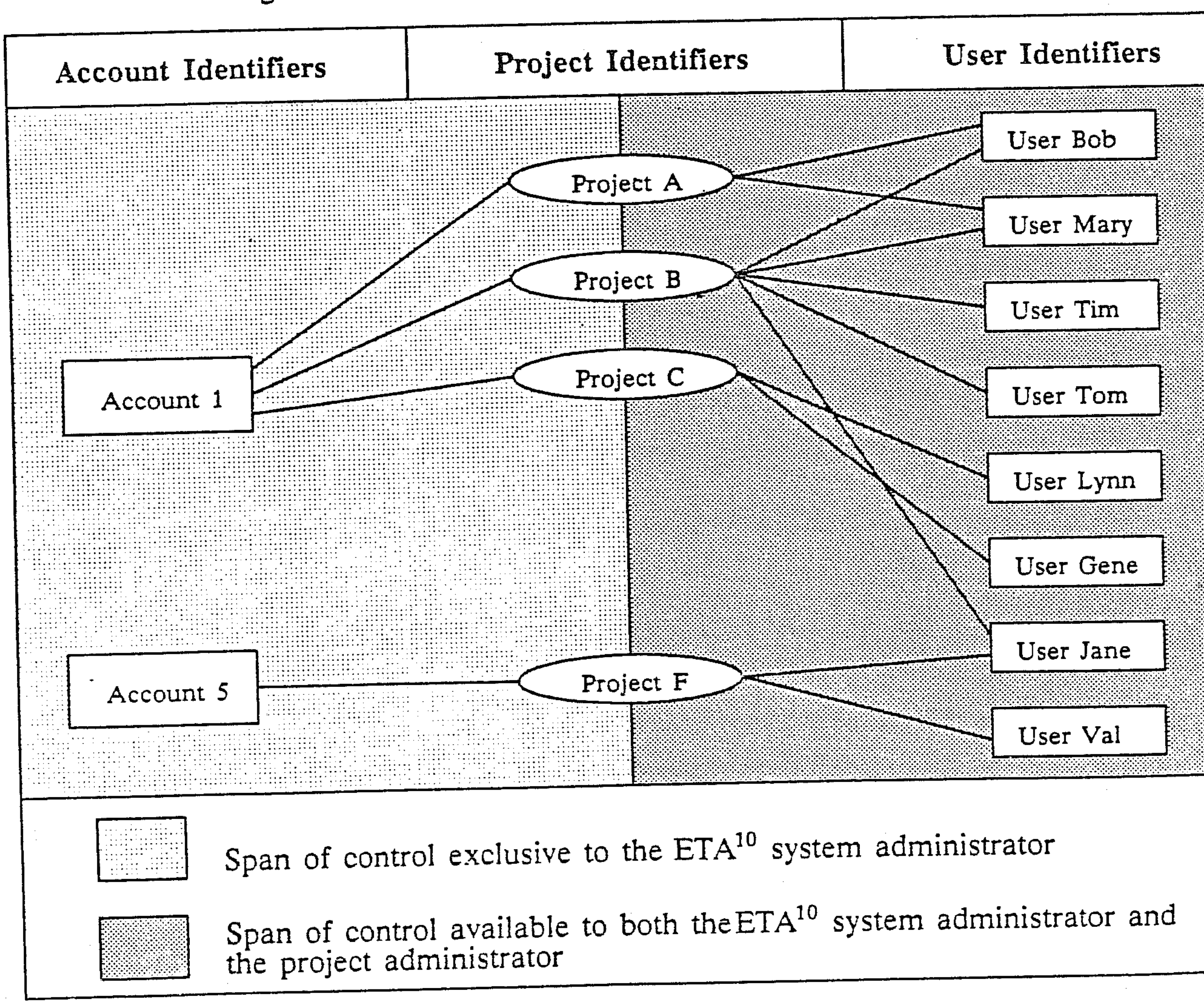


Figure 4–11. ETA10 system accounting hierarchy.

As illustrated in figure 4–11, the system administrator has overall control of the accounting system, and the project administrator has control of a particular project within an account. The accounting system is set up this way in order to accommodate installations that need more than one level of control for system resources.

Accounting and resource usage information for a particular project within an account is available to that project's administrator. Project information also is available to the system administrator, who receives all global accounting and resource usage information.

Resource Control

The ETA10 system regulates the assignment and use of all its resources (CPU cycles, memory, disk space, etc.). Accountable resources on the ETA10 are apportioned to accounts and projects. When user sessions or jobs run under a particular account/project pair, the resources are charged to that account/project combination.

User Access to the System

On the ETA10, system administrators maintain a user profile for each authorized user of the system. The user profile contains a list of which system features and resources that user can access. The user profile contains other data about the user, as well, such as default charging information, default file system working directory information, default user environment, and job control sequences that are executed whenever the user logs in or out. A user's profile can be augmented, changed, or deleted as required to meet sites' needs.

General Function and Distribution of the ETA10 Software

The kernel level of the ETA10 software is distributed across the entir hardware system. There are hardware-specific portions in each of th three processor types, Central Processing Unit, Input/Output Unit, an Service Unit (the processors are described in more detail in the next section), but most of the kernel-level software is independent of an individual processor. The UNIX and VSOS programming environments and some system administrator software run in the CPUs; the operator interface and the rest of system administration are the environments that run on the SU. Figure 4–12 shows the distribution of the underlying software system across the entire ETA10 system. The hardware architecture is described in section 5, *Hardware Components*.

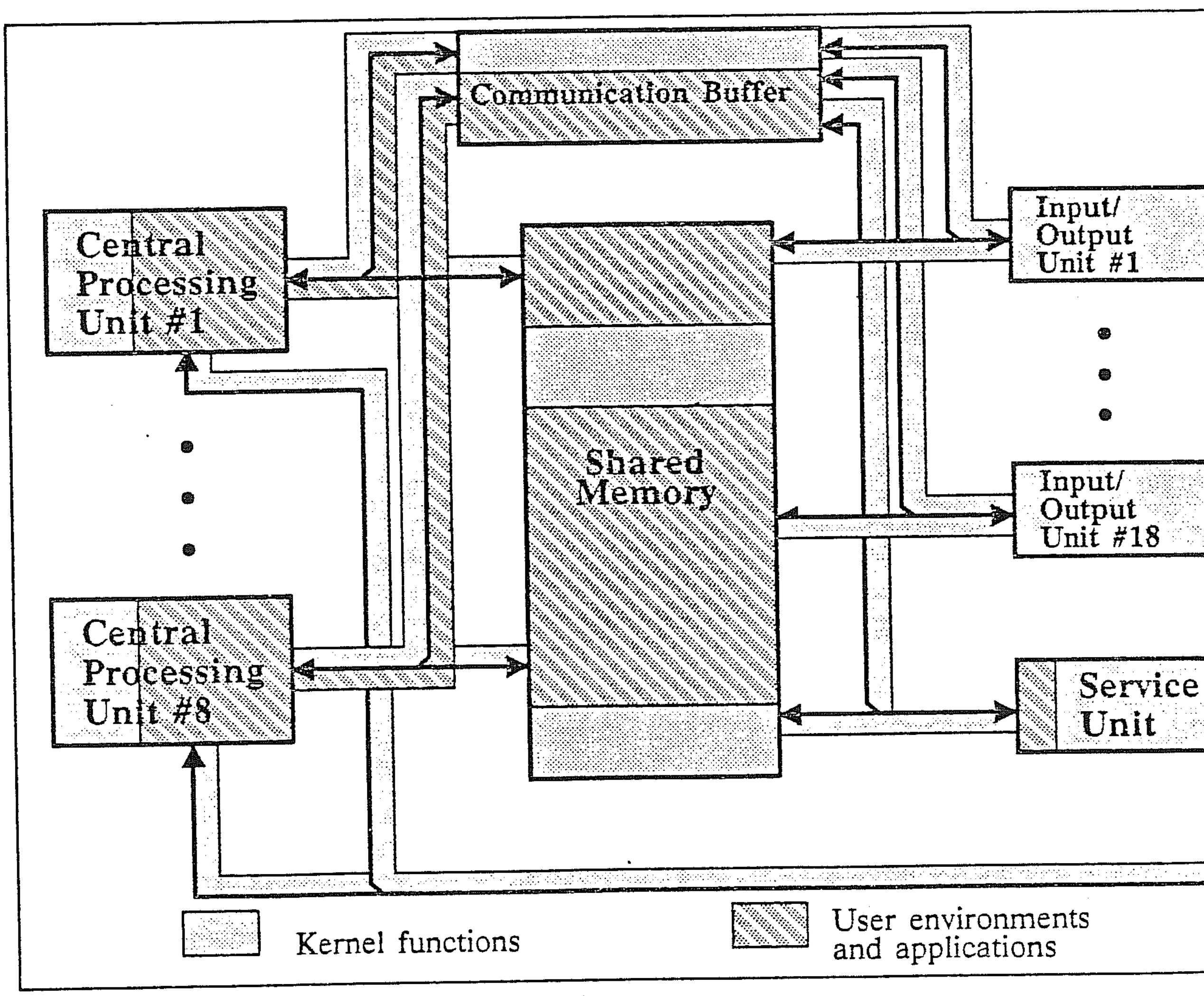


Figure 4–12. Physical distribution of the operating system across the ETA10.

Section 5

Hardware Components

Introduction

Several functional units, consisting of processors and their internal storage, provide the processing capability of the ETA10 system. The major hardware components of the system are as follows:

Central Processing Unit (CPU)	The combination of Central Processor and Central Processor Memory that is the computational engine in the system.
Input/Output Unit (IOU)	The collection of I/O Processors and channels connecting a data pipe to peripheral devices and the networks.
Shared Memory (SM)	The shared memory that provides file buffering and very high data transfer rates.
Communication Buffer (CB)	The communication buffer that acts as a low overhead mechanism for interprocessor communication.
Service Unit (SU)	The collection of maintenance and supervisory processors with dedicated peripherals, and a Maintenance Interface that provides data and control paths from the Service Unit to the other hardware components of the system.

Figure 5–1 illustrates the functional units of the ETA10 and their relationships to each other.

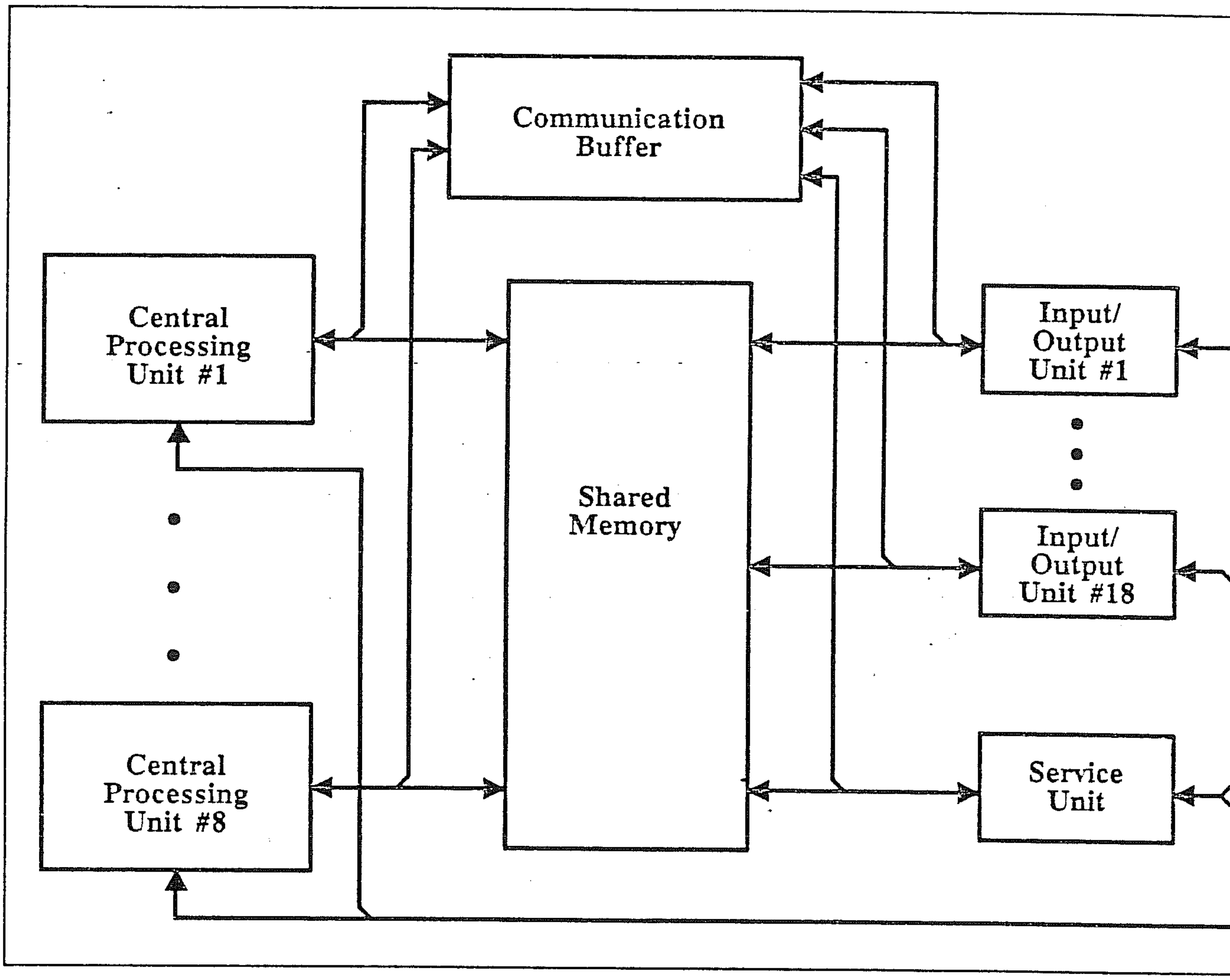


Figure 5–1. ETA10 functional diagram.

As shown in the above figure, the ETA10 is a multiprocessor computer system, with all processors having access to the large Shared Memory and the Communication Buffer. The ETA10 directly connects to high-speed disk and tape, and relies on network interfaces to a variety of computer systems, terminals, and workstations to perform low-speed I/O, introduce jobs, generate output, and support other peripheral equipment. All peripheral and network connections are through IOUs into Shared Memory.

Each ETA10 system includes a Service Unit, which provides for operator display and control, system reconfiguration, maintenance functions, and removable media for software installation.

Functional System Elements

The following subsections describe each of the major hardware components.

Central Processing Unit

The CPUs provide the real horsepower of the ETA10 supercomputer. Each Central Processor of the ETA10 includes a scalar processor, a vector processor, a large register file, and interfaces to other system components. Each Central Processor fits on a single printed circuit board, and operates at a normal temperature of less than 100° K for enhanced performance. Each CPU is capable of issuing an instruction every clock cycle. The Central Processor of the ETA10 is instruction and data compatible with the CYBER 205, and has several new instructions for accessing the shared memories in the system.

Other CPU characteristics are:

- 64-bit word
- two's complement arithmetic
- bit, byte, half-word, and full-word operations
- high-speed, 256-word register file
- instruction stack
- virtual memory with 1024-, 2048-, or 8192-word small page sizes; and 65,536- or 262,144-word large page sizes

The Central Processor Memory of each ETA10 CPU has a fixed size of four million virtually addressed words. CPM is constructed from 64K MOS Static Random Access Memory. The CPM bandwidth is 73 billion bits per second. Single error correction/double error detection (SECDED) protection is provided on each 32-bit half word. Access to Central Processor Memory is limited to the CPU with which it is associated, the maintenance port, and the shared memory port. The CPU diagram is shown in figure 5-2.

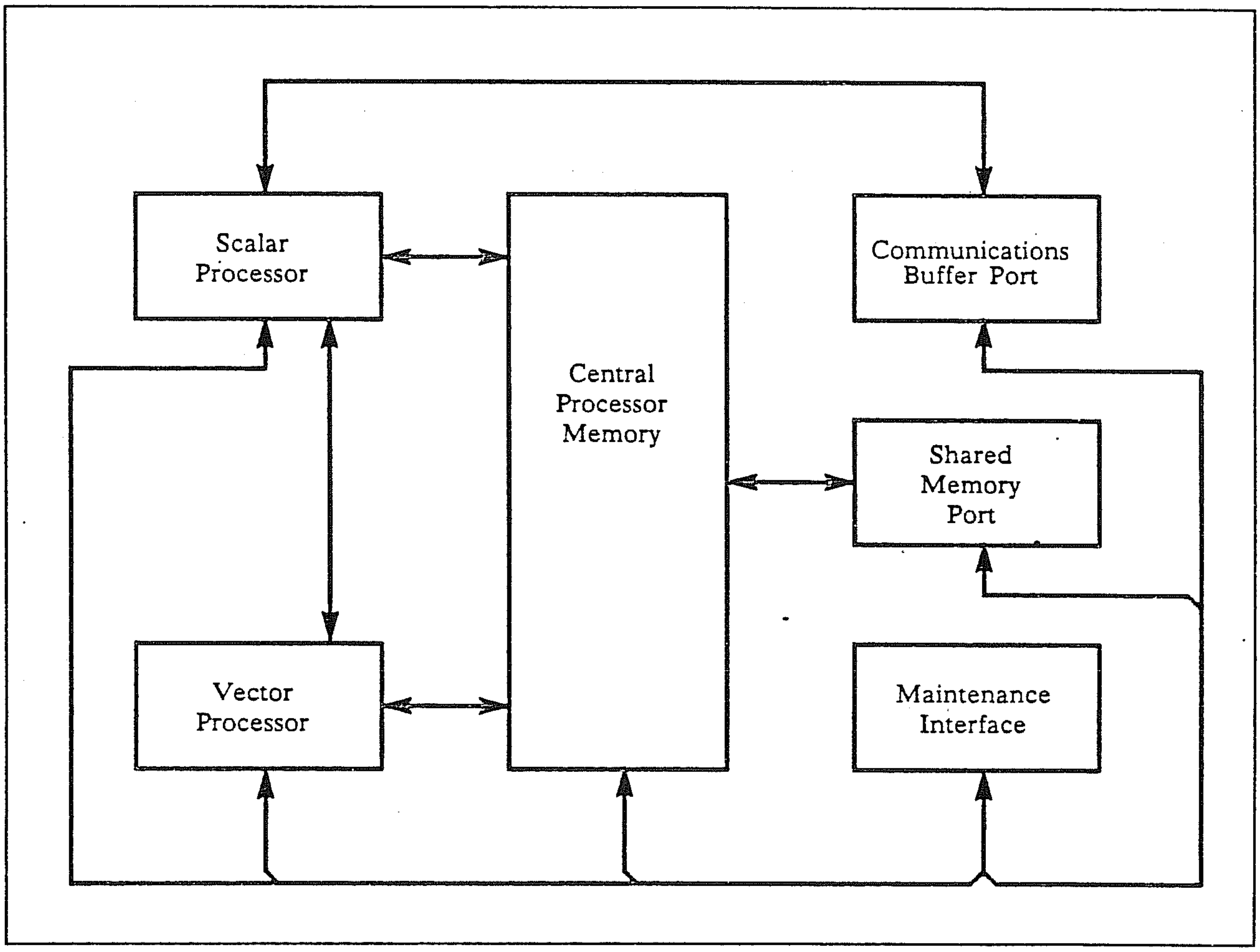


Figure 5-2. ETA10 Central Processing Unit.

The Shared Memory port and Communications Buffer port provide hardware interfaces from the CPU to each of those memories. The Maintenance Interface allows the Service Unit to connect directly to the CPU to perform diagnostic and maintenance functions.

Input/Output Unit

The Input/Output Unit provides the means to attach peripherals and networks to the ETA10 system. The base IOU consists of a cabinet, power supply, cooling, buses, maintenance interface, and Data Pipe Controller. Functional units that present a specific peripheral connection, or that provide a processor and memory increment, may be added to the base IOU. Each of these functional units provides a connection to disks, tapes, or a network. An IOU will hold up to eight functional units. All functional units in a single IOU share a Data Pipe to a Shared Memory low-speed port. Of the 20 low-speed Shared Memory ports, up to 18 may be assigned to IOUs. Low-speed port characteristics are discussed in greater detail in the next subsection. Figure 5-3 shows an internal diagram of the IOU.

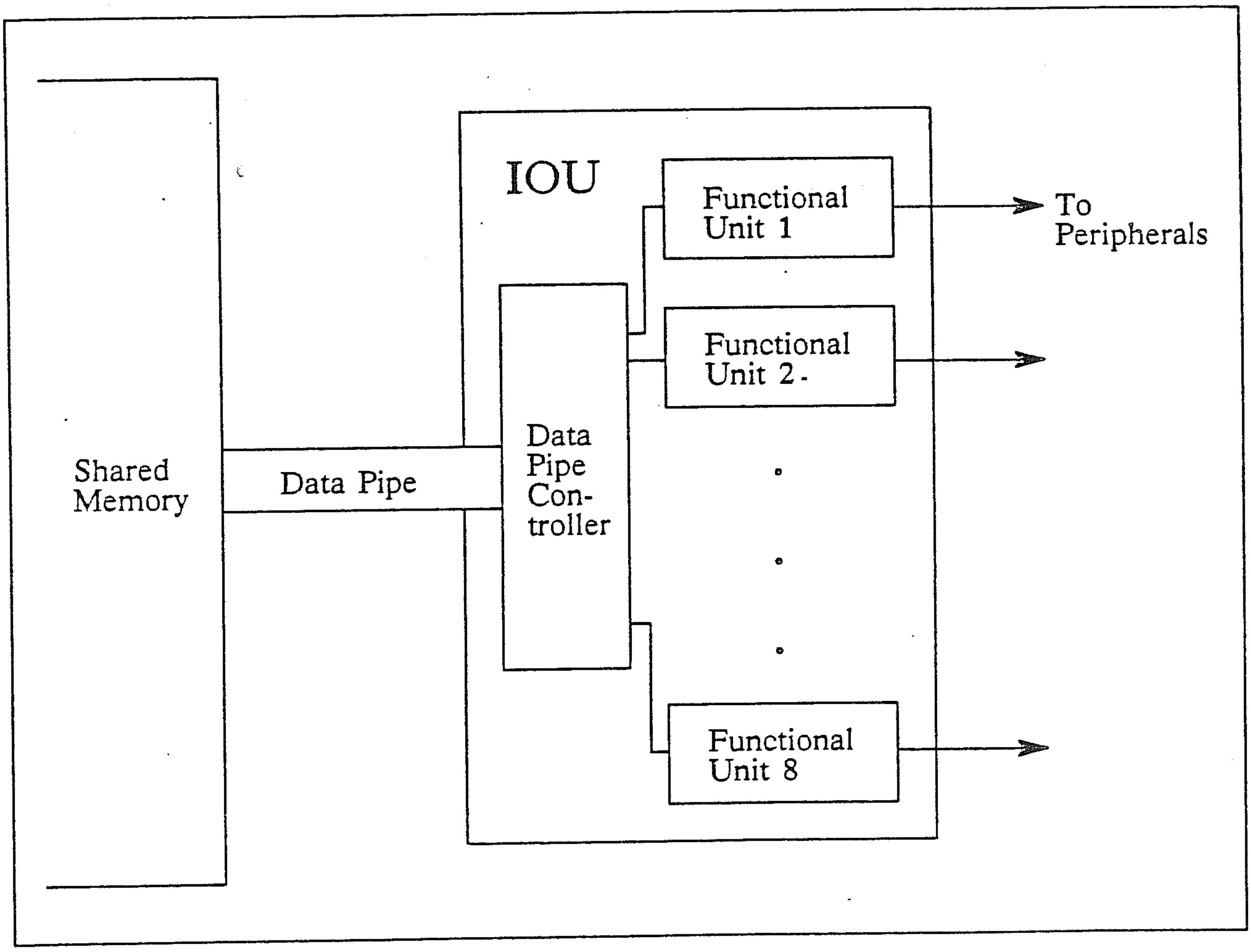


Figure 5-3. Internal diagram of the IOU.

Shared Memory Unit

The primary role of Shared Memory is to provide a large, high speed storage facility for active files. Shared Memory ranges in size from 64 million to 256 million words, in increments of 64 million words. Shared Memory is a 256K MOS Dynamic Random Access Memory. The Shared Memory Unit provides a total of 8 high-speed ports for CPU connection, with a sustained transfer rate of several billion bits per second per port; and 20 low-speed ports for IOU/SU connection with a sustained transfer rate of 350 million bits per second per port. The transfer size for a single request ranges from a half word to 65,536 words. SECDED protection at the 32-bit half word level is supported in Shared Memory. The system can be configured with degradable hardware so that during maintenance the Shared Memory Unit retains at least 50% of its normal operating capacity.

Communication Buffer

The Communication Buffer provides fast access to small shared data and synchronizing functions. The Communication Buffer has a fixed size of one million words. The Communication Buffer is constructed from 64K MOS Static Random Access Memory. The Communication Buffer bandwidth is 9.1 billion bits per second. The Communication Buffer provides unambiguous mutual exclusion operations for interprocessor synchronization. Access to the Communication Buffer and all its functions is available to all Central Processing Units, Input/Output Processors, and the Service Unit. The transfer size for a single request is a half word or a word. The Communication Buffer includes SECDED protection at the 32-bit half word level.

Service Unit

The Service Unit provides the ETA10 with operator consoles and system support tools. The Service Unit is a single network computer made up of two server nodes and two color graphics operator nodes. Up to six more (eight total) color graphics nodes may be attached to the network in order to provide additional operator consoles. These nodes are complete computer systems in themselves, and communicate with each other on a 12 million bit per second network. The SU also includes its own independent mass storage, which is shared by all nodes in its network.

The operator nodes connect to ETA Systems' Remote System Support Center by means of an attached modem. See section 7, *Customer Support and Training*, for more information about the Remote System Support Center. The modem supports 9600 baud communication over dial-up lines, and includes auto-dial and answer capabilities.

The Service Unit has the following peripherals: two 500M byte disk storage devices, one on each server node; one 86M byte Winchester disk with a built-in 62M byte, 1/4 inch streaming cartridge tape on each operator node; and one graphics printer.

The SU also includes a Power and Cooling Supervisor that collects and analyzes data about the power and cooling subsystems of the ETA10. In order to guarantee the safe and proper operation of the ETA10, the Power and Cooling Supervisor monitors and controls voltages, currents, pressures, temperatures, and other conditions.

Peripherals

The following subsections explain some of the performance characteristics of peripherals attached to the ETA10 system.

Disks

Device capacity for a single physical device on the ETA10 system is either 600 million bytes or 1.25 billion bytes. The ETA10 uses an Intelligent Standard Interface (ISI) to disk drives. ETA10 disks have a burst transfer rate of 12 million bytes per second, and a sustained transfer rate of 10 million bytes per second.

Tapes

Tapes supported on the ETA10 are 9-track, 1600 cpi Phase Encoded; and 9-track, 6250 cpi Group Code Recorded. Tape devices or their controllers interface to a FIPS 60 channel and comply with the FIPS 62 Standard for Tape Interface. On the ETA10, tapes provide dual channel access, the ability to connect up to 8 drives per channel, and a tape speed of 200 inches per second.

Networks

The ETA10 interfaces with both the Open Interconnection Network (OIN) and the Loosely Coupled Network (LCN), as described in section 3, *Networks and the ETA10 System*. The ETA10 also supports the collection and distribution of I/O between terminals and processes. It provides flow control for data and buffering of network I/O in concert with process scheduling.

Reconfiguring the ETA10

The ETA10 does not have to be restarted to be reconfigured around system components and data paths, including individual Central Processing Units, part of Shared Memory, individual Input/Output Units, the Service Unit, and the data paths between them.

All peripherals have at least two available access paths. The access can be switched dynamically from one path to another if needed.

For most hardware components, concurrent maintenance may be conducted on an inactive component without decreasing overall system capacity by more than 50%. Configuration management is controlled from the Service Unit.

Section 6

Documentation for the ETA[10] System

Introduction

ETA Systems recognizes the inherent differences in information needs between a system administrator, an operator, a programmer, and an end application user, and has organized the system documentation accordingly. Each type of user receives documentation that is appropriately detailed, organized, and worded for his or her needs.

This section describes the functional categories of the ETA10 user documentation and the types of documents found in each functional category. A map of the ETA10 user documentation appears at the end of this section.

Functional Categories of User Documents

There are two functional categories of ETA10 user documents:

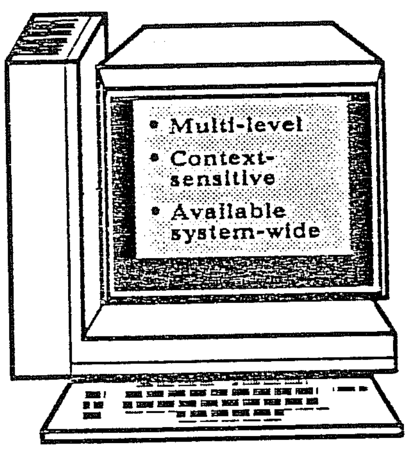
- system overview
- system use

System overview documents, such as this one, give a general description of a product or system. System use documents discuss topics such as the programming environments and specific application packages.

Types of Documents within Each Functional Category

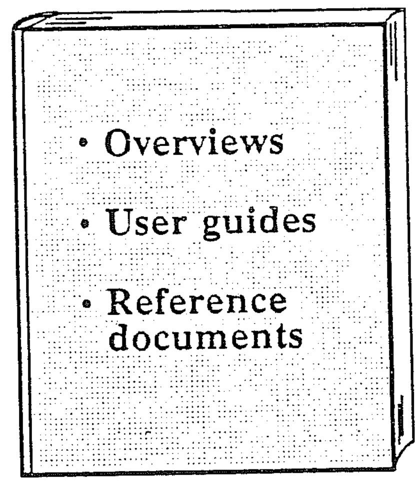
Four types of documents are included in the set of ETA10 user documentation. The four types are as follows: (1) on-line Helps, (2) overviews, (3) user guides, and (4) reference documents. The on-line Helps are available to interactive users on the ETA10 system. The other three types of documents are printed.

On-line Helps



On-line Helps are available to users of the programming environments and specific applications. When a user encounters an error on the system, he or she automatically receives a brief Help message. This initial Help gives users the minimal information they need to respond to the errors. Users who want more information can request further details about the error. This use of on-line assistance is context-sensitive. In other words, it considers the reader's context at the time Help is delivered, and customizes the assistance to the user. For instance, the user's file name, rather than some generic file name, may be included in an example. Users also can request more extensive Help about specific topics, without using the context of an error. This type of on-line assistance will be less context-sensitive; it will use generic file names, for example, but still is convenient for the user. When finished with either type of Help, users are returned to the points in their work where they left off.

Overviews introduce readers to a new topic. They provide readers with relatively short, general summaries of subjects. Emphasis is on broad, rather than deep, coverage of a topic.



User guides describe products or procedures in easy-to-read, step-by-step approach. They cover the most frequently used features of a given product and other products often associated and used with it. User guides usually focus on helping a user perform a related set of tasks or general function, rather than on the details of one product. For example, the *ETA FORTRAN User's Guide* describes how to create and debug FORTRAN programs, as well as how to use the most popular features of the ETA FORTRAN compiler. User guides describe concepts in general terms, avoiding the inner workings of their subject matter. Some user guides may be used as training material.

Reference documents contain complete, detailed, functional descriptions of products or procedures. They are organized by product, not by user tasks. While reference documents can be used with training materials, teaching readers how to use products or perform procedures is not their primary purpose.

Combining the Categories and Types of Documents

Figure 6–1 is a map that shows the system overview and system use portion of the ETA10 documentation. Each box in the figure includes one of the categories of documents described above. The various types of documents are listed within each of the categories. Specific titles are shown when possible.

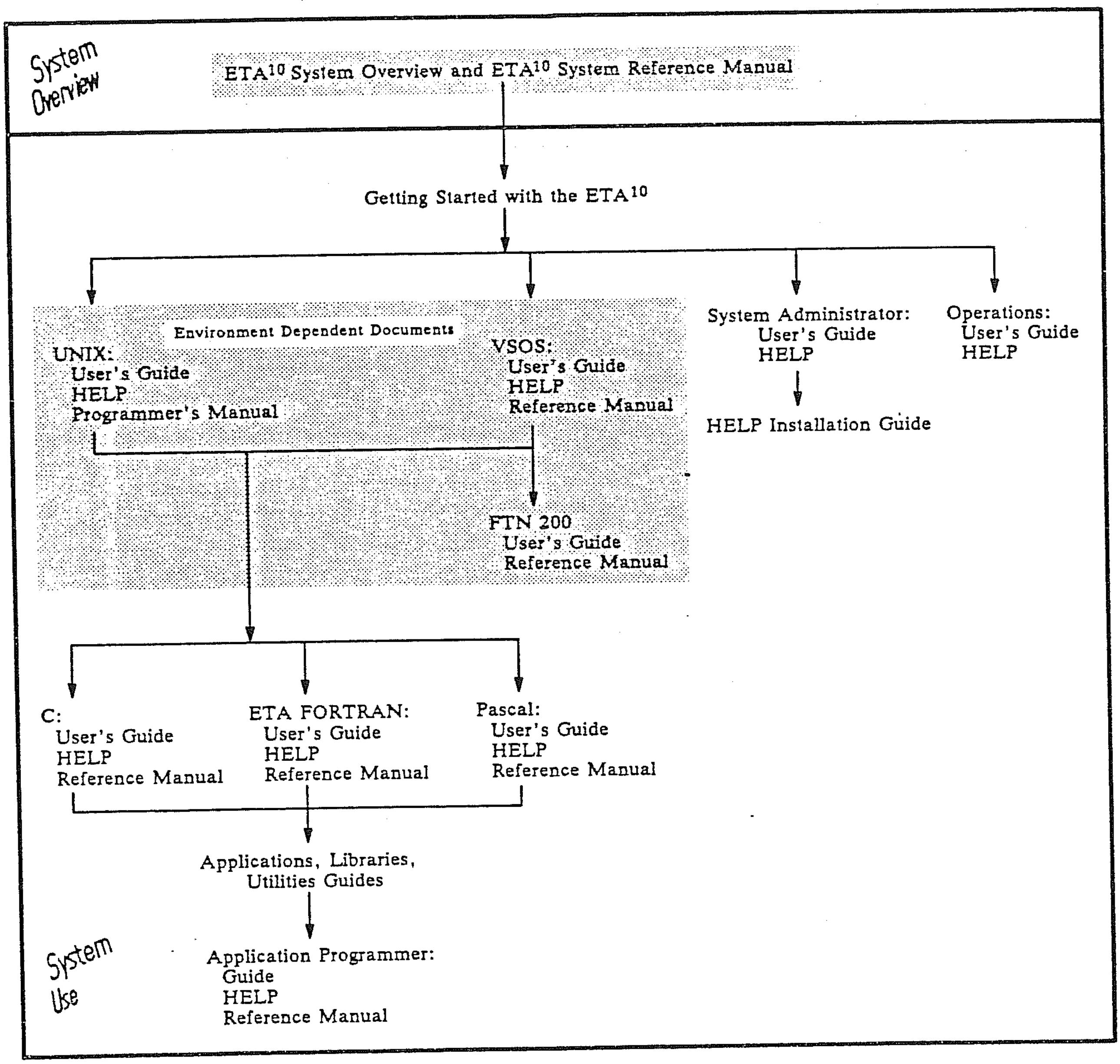


Figure 6–1. Map of system overview and system use portion of ETA10 documentation.

Section 7

Customer Support and Training

Introduction

This section discusses ETA Systems' customer support program and goals. Next it talks about the ETA10 software distribution, installation, and support. Finally, it discusses the training available for ETA10 customers.

Customer Support Program

ETA Systems' standard customer support is based on its Remote System Support (RSS) Center. The RSS Center is staffed by a group of highly-skilled system support specialists with access to a wide range of computer systems, telecommunications equipment, and data base facilities. Among other things, the RSS Center provides customers with 24-hour per day, seven days per week hot-line support. Technical assistance personnel and spare parts are dispatched by the RSS Center staff, as well.

Customer site staffing provided by ETA Systems includes a System Support Specialist. A selection of commonly needed spare parts is kept at the customer site, as well. During hours when System Support Specialists are not at the customer site, maintenance actions are performed by on-call personnel via the RSS Center.

In the future, ETA Systems will offer other levels of support that will reduce the number of ETA Systems personnel at the customer site. One result of this will be decreased cost for the customer.

The modular design of both hardware and software components of the ETA10 allows for graceful degradation of the system if necessary. System support personnel interact with all the hardware, software, and power and cooling system maintenance tools by means of a maintenance interface on the SU. If needed, RSS can perform maintenance activities remotely by using a high-speed connection to the customer site's Service Unit.

Remedial maintenance can be performed while the system continues to operate in a degraded mode. If there is a suspected hardware logic problem, system support personnel can use an on-chip maintenance system, Built-in Evaluation and Self-test (B.E.S.T.), to isolate the problem. Software support tools interface to an operating system debugger to aid in tracing errors in the system.

Software Distribution, Installation, and Support

Each ETA10 site receives a set of pre-built, binary code tailored for the system configuration. Tools are provided to install the software quickly. Part of these tools is a package that the system administrator uses to tailor the system for the site. With this tool, the system administrator defines the system workloads, then the system interprets this input and turns it into the appropriate tuning parameters.

ETA Systems supports the effective use of the system, as well as its installation. During the first two years of ETA10 system operation, there is one application analyst at the customer site. The application analyst has experience in the appropriate application areas to help the customer select effective algorithms for the ETA10. This also may include vectorizing applications and partitioning them for multitasking.

Training

Training for users of the ETA10 focuses heavily on system usage. The topics are organized into fairly small modules so students can take best advantage of class time. They can be trained on the specific topics they need to know for their jobs without having to cover material not related to them.

Because customer training is coordinated with the standard ETA10 manuals and on-line assistance, customers taking courses become familiar with the related ETA10 documentation.

Core courses for the ETA10 customers are designed in modules so individual customers can take as many or as few courses as necessary to meet their specific needs. The core curriculum for ETA10 customers includes the following courses:

- Introduction to the ETA10 (a two-part course: (1) System Overview, and (2) Getting Started)
- ETA10 Programming Environments
- Using the ETA FORTRAN Programmer's Tools
- System Administration
- Operator Training
- High Performance Programming
- How to Use Remote System Support (RSS)

Figure 7–1 illustrates the typical training paths for different types of ETA10 customers.

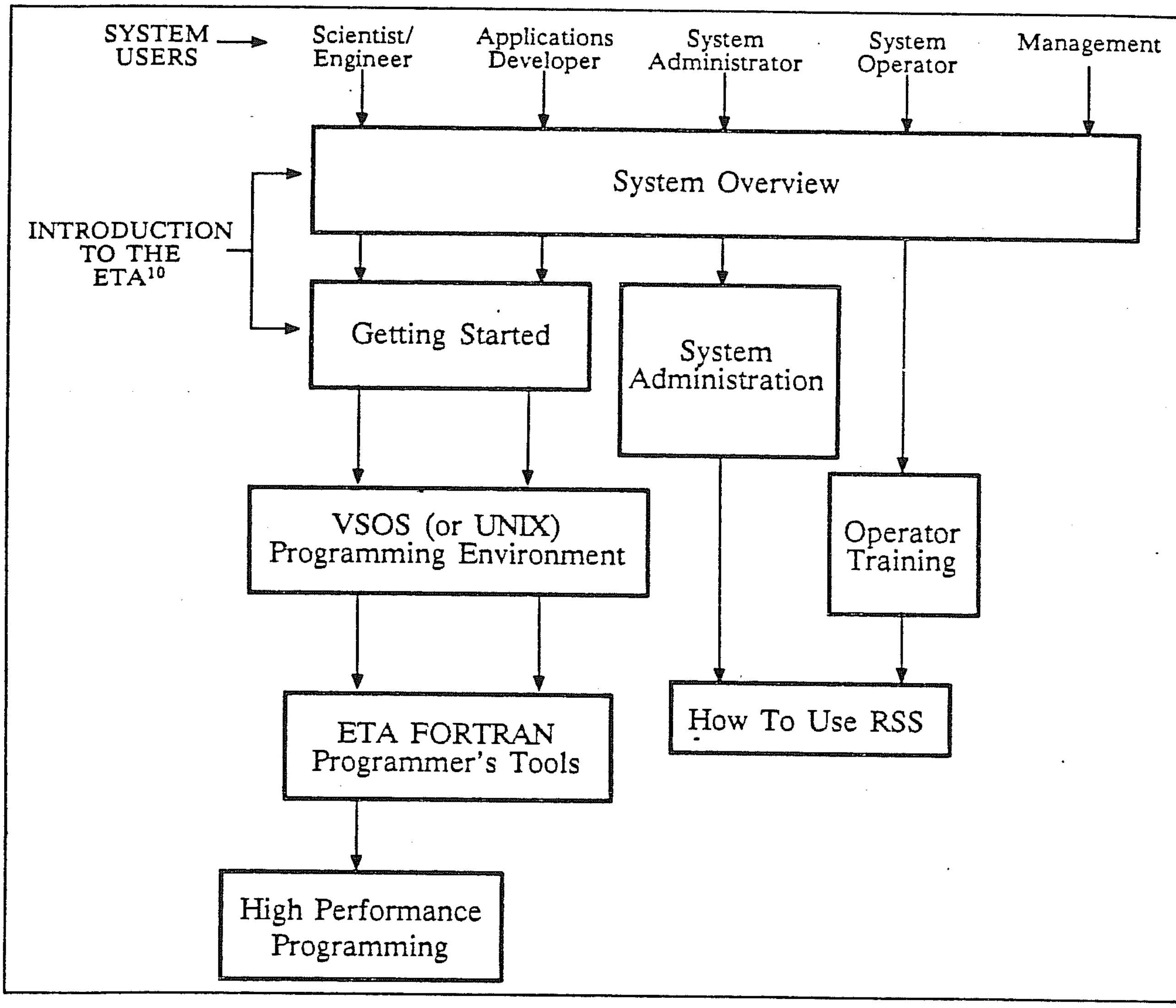


Figure 7–1. Training paths through core curriculum for ETA10 customers.

All courses are developed by ETA Systems and taught by specialized, experienced instructors. New courses will be added as the need arises.

One complete sequence of courses is taught on-site when a customer first receives an ETA10 system. After that, classes are offered at regular intervals in order to accommodate new users. Additional informal training is available from the on-site system analysts.

Section 8

For More Information

Introduction

This document has presented a broad perspective of the ETA10 System and described some of its individual parts. More information about these topics is being developed now; some of these documents are described in this section.

Site Manual

Getting Started with the ETA10 System will introduce the support staff, programming enviroments, hardware, and information available at each ETA10 installation.

Programming Environments

The *VSOS User's Guide* will acquaint programmers with VSOS features, commands, and tools. Similarly, the *Introduction to the UNIX Environment* will describe standard UNIX features, introduce user commands and programming tools, and point the way to other UNIX documents.

Programming Languages

The *ETA FORTRAN* and *FORTRAN 200 User's Guides* will explain how to develop programs and use the programming tools available in each environment. The *Introduction to the C Language* will introduce C constructs, libraries, compilers, and support tools.

Acronym Guide and Glossary

ACRONYMS

ANSI	American National Standards Institute
ASCII	American Standard Code for Information Interchange
AT&T	American Telephone and Telegraph
B.E.S.T.	Built-in Evaluation and Self-test
CB	Communication Buffer
CDC	Control Data Corporation
CPU	Central Processing Unit
DEC	Digital Equipment Corporation
EBCDIC	Extended Binary Coded Decimal Interchange Code
FIPS	Federal Information Processing Standard
FORTRAN	Formula Translation (programming language)
FTP	File Transfer Protocol
IBM	International Business Machines Corporation
IEEE	Institute of Electrical and Electronics Engineers
I/O	Input/Output
IOI	Input/Output Interface
IOP	Input/Output Processor
IOU	Input/Output Unit
ISI	Intelligent Standard Interface
LCN	Loosely Coupled Network
MVS	Multiple Virtual Storage Operating System (IBM)
NOS	Network Operating System (CDC)
RHF	Remote Host Facility (CDC)
RSS	Remote System Support (Center)
SECDED	Single-bit Error Correction Double-bit Error Detection
SIL	System Interface Language (of VSOS)

SNA	System Network Architecture
SU	Service Unit
TCP/IP	Transmission Control Protocol/Internet Protocol
VME	VERSA Module Eurocard
VMS	Virtual Memory System (DEC)
VSOS	Virtual Storage Operating System (CDC)

GLOSSARY

Applications	User-accessible software designed to perform a certain function or solve a particular type of problem when given appropriate commands and data by the user.
Central Processing Unit (CPU)	The central processor, central processor memory, and associated cooling systems and power supplies.
Central Processor (CP)	The scalar processor, vector processor, register file, and interfaces to other system components.
Central Processor Memory (CPM)	Memory that is associated with one CP. It also is accessible from SM and the Service Unit.
Channel	The interface between I/O processors and peripheral devices (disks, tapes, or various networks).
Communication Buffer (CB)	A memory, accessible from all processors of the system, used for transmission of high speed synchronization messages and signals throughout the system components.
Data Pipe (DP)	The connection between the Shared Memory and an IOU. This connection is shared by all functional units and I/O channels in the IOU.

Documentation Documents that support computer hardware, computer software, computer systems, and services relative to them.

Gigaflop One billion floating-point operations.

Input/Output Unit (IOU) A collection of I/O processors and channels that connect a data pipe to peripheral devices and networks.

Job A job is a user-defined set of processes (serial or concurrent) and system commands. In addition to being a set of actions, the job is an environment that provides for the execution of the processes and commands.

Kernel The basic, general software functions that manage the ETA10 hardware and provide services for the programming environments. It is organized into features, with each managing a hardware resource, or supporting the operation and structure of the software system. Parts of the kernel run in all system processor types.

Loosely Coupled Network (LCN) A network used for communications between one host mainframe and another.

Maintenance Interface (MI) The combination of hardware and software that provides all connectivity and buffering between the system components and the Service Units.

Multi-processor A computer that has two or more processing units which have access to a common main storage.

Multi-programming A mode of operation that provides for the interleaved execution of two or more computer programs by a single processor.

Multitasking A mode of operation that provides for the concurrent execution or interleaved execution of two or more tasks.

Network — A system of interconnected processing entities, which may include computers, I/O devices, and storage devices.

Network Access Device (NAD) — Device that provides for connection and access to a Loosely Coupled Network.

On-line Help — An interactive system that provides a range of assistance from simple command assistance to elaborate and detailed tutoring and command reference.

Page — An allocation unit of CPM. Two page sizes (one large page and one small page) are used at one time on each ETA10 CPU.

Page Fault — An interrupt that occurs when a process references a page that does not have an entry in the page table, or when the process does not have the correct capability for a particular page access.

Page Table — A table that contains the page table entries. The hardware uses a copy of the page table to associate a virtual address with a physical memory address.

Paging File — A disk file that contains modified pages of CPM that were swapped out of CPM.

Peripheral — A data storage device, or a unit on a network, that is electrically connected to the mainframe.

Process — A sequence of operations that produce a specified result. On the ETA10 system, a process is the smallest unit of program execution and resource assignment.

Processor — A functional unit that interprets and executes instructions in a system. *See* Vector Processor, Scalar Processor, Central Processor, *and* Input/Output Processor.

Programming Environment The environment in which a job executes. An environment typically consists of a command processor and a set of utilities, libraries, and languages. The programming environment defines the control language which directs job flow and dependencies, resource assignment, and type of status information, if any, returned from control statement execution.

Protocol The rules and conventions that govern the interchange of (digital) information between two (point-to-point) or more (multipoint) entities.

Remote Host Facility (RHF) Software that provides host-to-host communication using the loosely coupled network (LCN).

Remote System Support (RSS) A central ETA Systems facility that supplies system support to customers.

Scalar Processing Performing operations on quantities characterized by single values. (*Contrast with* Vector Processing.)

Scalar Processor The unit that performs scalar processing.

Serial Processing The sequential or consecutive execution of two or more processes in a single device.

Service Unit (SU) The hardware and software that provide system monitoring capabilities and the operator interface.

Shared Memory (SM) A large memory that is available from all processors in a system. SM is used for fast access to file data.

Synchronous When two or more processes depend on the occurrence of a specific event such as a common timing signal.

System	1. Software: The collection of methods, procedures, software, and techniques united by regulated interaction to form an organized whole in order to satisfy a specific set of requirements. 2. Hardware: The configuration consisting of some mainframe subsystem, an I/O subsystem, and a service subsystem. 3. Overall: The combination of hardware, software, documentation, and support that make up the ETA10.
Task	In a multiprogramming or multiprocessing environment, one or more sequences of instructions treated by a control program as an element of work to be accomplished by a computer. The basic unit of work from the standpoint of a common control program.
Vector Processing	Performing operations on quantities usually characterized by an ordered set of scalars.
Vector Processor	The unit that performs vector processing.
Virtual Storage	Storage space regarded as addressable main storage by the user of a computer system. Virtual storage is mapped into physical storage.

Reader Comment Sheet

Please take a few moments to review this ETA Systems publication:

- How do you use this publication? (For example, do you read it to get a general understanding of the subject, or to look up specific information? How often do you use this document? With how many other people do you share this publication?)

- Is the material organized appropriately for your purposes? (If not, please describe a more appropriate organization of the material.)

- Are you able to locate needed information easily and quickly? (If not, what would you suggest to make it easier and quicker to locate information?)

- Is the text clear and concise? (Please identify any items that you feel are confusing, ambiguous, or too lengthy.)

- Is the material complete enough to meet your needs? (If not, what is missing?)

- How effectively are examples and illustrations used? What would make them better?

- Is the publication physically appropriate for your use? (For instance, is the publication the proper size? Are you able to arrange the publication for easy use? Is the publication durable enough?)

- Are there any errors in the publication? If so, please list them here, including page number references and descriptions of the needed corrections.

Please use this space for your general comments about ETA Systems publications.

Thank you for your time and input. Please fold, staple, and send to the address shown below.

fold 1

fold 2

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 2294 ST. PAUL, MN

POSTAGE WILL BE PAID BY ADDRESSEE

ETA SYSTEMS, INCORPORATED
TECHNICAL COMMUNICATION DEPT.
1450 ENERGY PARK DRIVE
ST. PAUL, MN 55108