Chapter 5

The Operating System: CMOST

The CM-5 operating system, CMOST, is an enhanced version of the UNIX operating system. The enhancements optimize computation, communication, and I/O performance within the CM-5 system itself, while the adherence to UNIX standards allows the CM-5 to interact efficiently with other computers in a heterogeneous, networked environment.

Because the CMOST operating system is built upon standard UNIX, it can provide all the services that any standard network server provides:

- timesharing and batch processing
- standard UNIX protection, security, and user interfaces
- support for all standard UNIX-based communications protocols
- exchange of data with other systems in an open, seamless fashion
- the ability to access files on other systems via NFS protocols and to supply data to other systems by acting as an NFS server
- the Network Queuing System (NQS) and other standard network-oriented programs
- for scalar programs, binary compatibility with SunOS

Enhancements provide higher-performance services and expanded functionality for users within the CM-5 system:

- high-speed file access
- fast parallel interprocessor communications capabilities
- other parallel operations for optimal utilization of CM-5 hardware

Copyright © 1993 Thinking Machines Corporation

- central administration and resource management for all CM-5 computational and I/O facilities
- support for extended models of data parallel programming, such as data parallel pipes
- support for other parallel programming models
- checkpointing

5.1 CMOST and the CM-5 Architecture

The computational nodes on a CM-5 are grouped into partitions. A partition can be as small as 32 processors, or as large as the entire machine. The partitioning is flexible and is controlled by the system administrator, who can create and alter partitions as needed to meet site requirements. Each partition operates independently under a control processor acting as a partition manager (PM). Users log in to (or **`rsh`** on to) the PM and, once logged in, have full access to the PM itself, to all the computational nodes it controls, and—through the operating system—to all the I/O resources, partitions, and network connections of the CM-5 system. Figure 13 shows a user's-eye view of the CM-5.

Each partition manager runs a full version of the CMOST operating system. The PM makes all operating system resource allocation decisions and all swapping decisions for its partition, as well as most system calls for process execution, memory management, and I/O.

Each processing node runs an operating system microkernel, which supports the mechanisms required to implement the policy decisions made in the partition manager. All operating system code operates in supervisor mode, allowing it to access any network address and memory address in the machine.

When a user process begins running, its partition manager downloads code to the processing nodes and broadcasts identical memory maps to each node. The nodes then execute the provided code, each acting on its own data and executing computations and branches accordingly.

All nodes in a partition operate on the same process at the same time. Interprocessor communication between nodes within an application is handled

November 1993
Copyright © 1993 Thinking Machines Corporation

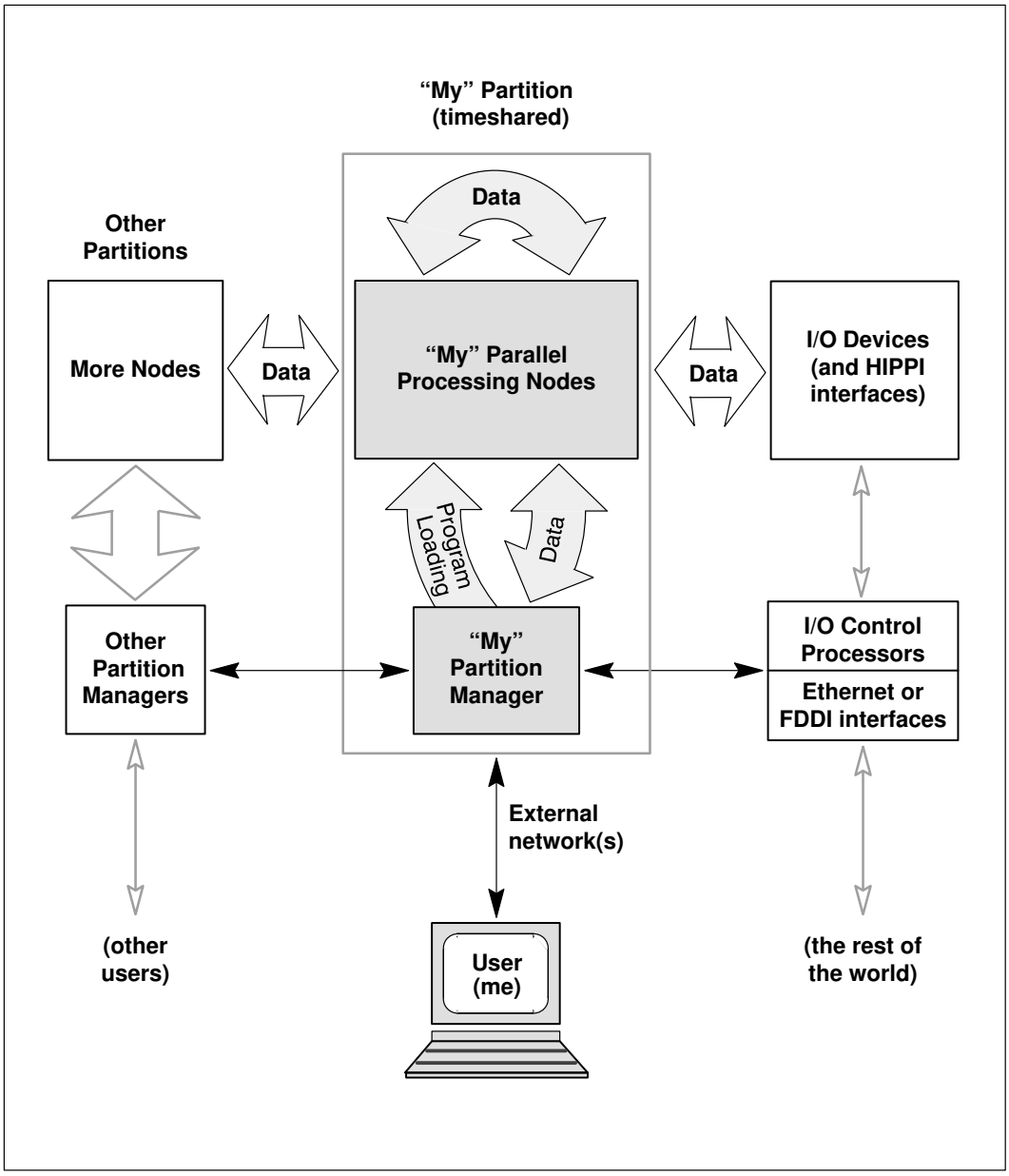

Figure 13. A user's view of a CM-5.

Users access a CM-5 system by running `rlogin` or `rsh` commands on a specific partition manager. A user program begins execution on the PM, downloads code to the nodes, then runs on nodes and PM both, passing data as needed among processors.

If a program needs to exchange data with an I/O device or with another process, the PM arranges the transfer, via system calls to other control processors. Data then flows directly between the nodes and the I/O device, nodes of another partition, or external network interface, thus ensuring that a parallel process gets the full benefit of the CM-5 Data Network bandwidth.

Copyright © 1993 Thinking Machines Corporation

entirely by user code, without any operating system overhead. For external communications the user process calls on the operating system, which requests and supervises the transfer on behalf of the user process. Data may be transferred between two processes running timeshared in the same partition or between two processes running concurrently in different partitions.

Interprocess communication is based on parallel extensions to UNIX sockets and pipes and is managed by the operating system. I/O transfers are handled in the same manner as transfers between partitions.

5.2 CMOST and the Users

Users typically access the CM-5 through an external network, either in batch mode, via the NQS **`qsub`** command, or interactively, via **`rlogin`** or **`rsh`** commands.

Each PM and IOCP within the CM-5 is a separate host on the network. Users can log in to any PM or IOCP for which they have appropriate privileges. Once logged in, a user has access to the full resources controlled by that control processor and to both local and networked file systems; the user can then run processes that use a control processor alone or a full partition of PM plus processing nodes. Since the set of control processors (PMs and IOCPs) within a CM-5 form a loosely coupled network of UNIX computers, a user with appropriate privileges can also run programs on any processor within the CM-5 using the normal UNIX networking commands.

The Program Development Environment

The program development environment available to CM users offers the full capabilities of UNIX and the X Window System. In addition, it offers enhancements specific to CM parallel programming: parallel languages, specialized libraries, and tools for parallel debugging and performance analysis. Prism, the CM-5's integrated programming environment, facilitates programmers' use of the machine (see Chapter 6).

November 1993
Copyright © 1993 Thinking Machines Corporation

The Program Execution Environment

The program execution environment on the CM-5 supports both interactive, time-shared program execution and batch execution using the NQS batch system.

Several facilities, such as automatic checkpointing and Prism, the CM programming environment, aid program development and robustness during execution. (See Chapter 7.)

5.3 CMOST and the Administrator

CMOST provides the administrator with tools for efficient and flexible resource management. It allows the administrator to partition the CM-5 for spacesharing among users, to set up the NQS batch system and the accounting system, and to monitor system usage, error logging, and power and environmental concerns. In addition, it provides all the standard UNIX capabilities, such as setting process priorities for use with process scheduling, setting disk quotas to control disk space usage, backing up and restoring user data, and setting up user permissions.

CM-5 administration is centralized at a system console, using commands that are modeled on SunOS 4.1 commands. The commands execute through a set of daemon processes that run (depending on their tasks) on the system console processor, the diagnostic console processor, or the partition managers.

5.4 I/O and File Systems

I/O programming on the CM-5 uses standard UNIX mechanisms, including sockets, pipes, character devices, block devices, and serial files. All I/O operations are modeled as reads and writes to files, regardless of the type of device used for storage.

CMOST extends the UNIX I/O environment to support parallel reads and writes and to support very large files, including files above the size supported in most current UNIX implementations. The virtual file system interface supports device-independent file behavior and supports many different file system types, including the standard UNIX file system, the Network File System (NFS), and two CM file systems: CMFS, which is supported on all Connection Machine systems and which allows the CM-5, the CM-2, and the CM-200 to share files, file

Copyright © 1993 Thinking Machines Corporation

systems, and I/O peripherals; and a high-performance file system, SFS, that is supported only on the CM-5.

The CM-5 arranges communications to allow maximum simultaneous performance of computation and I/O. Transfers from one partition do not affect the performance of other partitions. Simultaneous transfers from several partitions see minimal interactions unless they require access to the same I/O device. Direct I/O-to-I/O transfers allow direct movement of data between a remote machine and a CM-5 I/O device, or between primary and secondary I/O devices on a CM-5, without affecting activities in partitions.

The CM-5 File System

The Connection Machine file systems manage the CM's high-speed disk storage (Scalable Disk Arrays or DataVaults), and other I/O peripherals.

Within CM-5 files, data is stored in canonical (serial UNIX) ordering, thus allowing its use by both serial and parallel systems and processes. When a serial process does I/O, data remains in canonical order throughout; for parallel I/O, data moves between the canonical order and the ordering required by the computational nodes.

This reordering serves two important purposes. First, it allows a program to run on partitions of any size without affecting its I/O: a file written by a process running on a partition of one size may be read with equal ease by a process running on a partition of a different size. Second, it allows the same file to be read by parallel or serial processes. A serial process may read a file written by a parallel process, and vice versa.

For further information on the CM-5 file systems and I/O, see Chapter 20.

Network Communications

Data can travel through sockets directly between CM-5 processes and other machines on the network. A user process can create a socket, send parallel data to it, and have that data received as a serial stream by a serial or vector computer. The same socket can carry serial data from control processors; as with file I/O, network communication uses standard protocols and data ordering for transmission, and uses parallel ordering only within the parallel computational nodes.

November 1993
Copyright © 1993 Thinking Machines Corporation

The User's View

From the user's point of view, data from any file system, on any device, appears the same and is handled in the same manner. A CM-5 control processor, accessing data over the Data Network, sees no difference between data stored on any CM-5 I/O device and data stored on any other UNIX file system.

Similarly, user processes are not concerned with the storage media on the CM. Whether data is stored on a single device or striped across multiple devices, the process accesses it as a single file. The only user-visible difference is in performance.

Copyright © 1993 Thinking Machines Corporation