Management and Development Local Area Network Upgrade Prototype

T. J. Fouser
DSN Data Systems Section

Given the situation of having management and development users accessing a central computing facility and given the fact that these same users have the need for local computation and storage, the utilization of a commercially available networking system such as CP/NET from Digital Research provides the building blocks for communicating intelligent microsystems to file and print services. The major problems to be overcome in the implementation of such a network are the dearth of intelligent communication front-ends for the microcomputers and the lack of a rich set of management and software development tools.

I. Introduction

The purpose of this paper is to report on the progress of a research effort to study local area networks and the application of networking to administrative and developmental needs. A local area network of communicating, intelligent workstations provides the users with enough localized computing power to perform tasks such as word processing, program development, and other applicable implementation engineering work. The network with a particular node designated as the server, equipped with extra disk storage and one or more printers of different capabilities, provides the users with print server and file server functions. The research effort is directed toward investigation of commercially available microcomputer operating systems, CP/NET, CP/M, and MP/M.[1] CP/M has become the industry-wide standard microcomputer operating system. MP/M is the multi-user, multiprogramming version of CP/M, and CP/NET is the networking interface between a user and his CP/M system and the server node operating with MP/M. CP/M, CP/NET, and MP/M are products and trademarks of Digital Research.

[1]CP/NET, CP/M, and MP/M are trademarks of Digital Research Inc., Pacific Grove, Calif.

II. Background

The current configuration of the Management and Development Network (MADNET) is illustrated by the top portion of Fig. 1. The system centers around dual MODCOMP 7870 computers, one of which supports the HAL/S software data base, three printers, and up to 26 users of the HAL/S software development effort. The other supports the work breakdown structure (WBS) data base, two printers and 28 management users. Each of the users has a MADNET standard, DEC VT-100 terminal and runs the WBS software on the 7870, generating reports from the data base that are output to the printers. The VT-100 terminals were chosen since they could display 132 columns of output when generating reports and are considered to be a constant in the upgrade process. With the existence of many VT-100 terminals which can be upgraded to CP/M personal computers with a commercially available product and with the existence of many AODC computers, networking these systems together was primarily concerned with bringing up a server node with the desired peripherals and fine tuning the network software for optimum performance.

III. Implementation Effort

The lower portion of Fig. 1 indicates the added CP/NET system. The dumb VT-100 terminals on the MODCOMP are upgraded with the addition of a Z80A cpu, 64-kbytes RAM, floppy disk controller, parallel printer port and four serial ports. These components are on two cards that are housed inside the VT-100 cabinet and connected to one or more floppy disk drives in a separate enclosure. The upgraded terminal runs CP/M at 4 MHz and allows for dumb terminal emulation when needed for operating with the MODCOMP. The upgraded terminal can have an optional printer that serves as the CP/M list device. Communication to the network is provided through the additional serial I/O ports.

The central CP/NET node, the server, in the current prototype studies, consists of a Z80-based microcomputer running MP/M along with the CP/NET server software. The hardware is a Z80A running at 4 MHz with 64-kbytes RAM, seven serial I/O ports, two single-sided single density floppy disk drives, and one Morrow 26MB hard disk. The seven serial ports support two local consoles for MP/M operations, one serial printer port, and four serial ports for network communication. The communication with the nodes can occur at various baud rates. Currently, direct connections run at 9600 baud and connections through modems run at 1200 baud.

The software on the CP/NET server, as shown in Fig. 2, consists of the standard MP/M system software and utilities with additional CP/NET utilities and a customized extended I/O system (XIOS). The MP/M multiprogramming monitor control program provides a microcomputer environment with multiple consoles, each with multiprogramming capabilities. The standard MP/M features include spooling of print files to the printer, scheduling of programs to be run by date and time, and setting and viewing the current date and time. Each user at a console has the ability to start a process, detach the console from that process, and initiate another process. The ability of MP/M to support detached processes allows processes that are always in memory acting as a resource. The CP/NET software executes in this manner, providing support for node operations such as printer and disk accesses. The CP/NET server software adds capability to receive messages from the nodes and send messages to the nodes, with the ability to broadcast to all nodes. The XIOS provides the custom interface between the standard MP/M and CP/NET software and the hardware the system is running on.

The software on the CP/NET node consists of the standard CP/M and the network disk operating system (NDOS), which intercepts operating system calls that need to be redirected onto the network. The customized slave network I/O system (SNIOS) interfaces the standard software to the particular hardware configuration and provides the communication over the network.

The mapping of the CP/NET configuration to the International Organization for Standardization's Reference Model of Open Systems Interconnection (ISO OSI) is shown in Fig. 3. The physical layer is the hardware that transfers the bits from one node to another without regard for whether or not the collections of bits constitute a valid packet. The data layer picks the checksum out of the transmitted bitstream and judges the integrity of the packet. These ISO Reference Model layers are implemented in the hardware and the customized XIOS in the server and SNIOS in the nodes. The network, transport, session, and presentation layers of the Model are implemented in the standard CP/NET software, the NDOS in the nodes and the node support processes in the server.

IV. Operation of Current Prototype

The operation of the CP/NET local area network provides the user with print and file server functions. A typical session might be retrieving a file from the file server, performing the desired operations on the file such as editing or other computing, perhaps printing the file or processing it with printed report output, then restoring the file to the file server. Once the print file is generated, it is then sent to the print server for spooling and printing.

The applications of these functions include electronic mail, common software, off-loading of the editing function from the MODCOMP, and WBS off-loading and viewing. The utilization of data base management systems on the server provides for operations such as SRM, ECM, WAD and action item accessing.

The use of the network as a print server allows operations to be performed with as little as possible impact on the user. The typical node has a slow and possibly noisy local printer or, more likely, has no printer at all. The local printer is used for print jobs that are short or do not require a printer with special features not found on the local printer. To use the print server, the local user first equates the CP/M print device with the desired printer on the server, then uses the print device as usual. A program such as Wordstar[2], generating printer output, would have that output intercepted by the NDOS and rerouted to the network and on to the server where it would be automatically spooled for printing. When the printing is complete, the end-of-printfile message is sent to the server, triggering the despooling of the print file. The local user also uses the CP/M file transfer utility to send previously generated print files to the print server, where they are automatically spooled for printing.

The additional capabilities of the CP/NET must be stated in terms of utility to the users. For use as a file server, the network must be able to store the user's files such that they can be retrieved without undue delays. Speeds of the current configuration do not approach those of using the local disk, discouraging the use of the file server for temporary storage. The ease of this operation to the user also depends on the configuration of the user's hardware. If the node has a local printer that is letter quality and/or slow, sending large print files to the server is faster than printing the file at an effective baud rate of 300 to 1200 bps. If the local printer is fast and can produce the required quality, use of the print server would be of questionable value. The cost of equipping each node with such a printer may be prohibitive.

To determine the actual performance of the network file services, a "typical" file size was selected. The file is slightly larger than 6 kbytes, which represents approximately four pages of typewritten material. Since a typical operation would include transferring from server to node, then node to server, both times were investigated. The nature of the environments of the server and the node made a significant difference in the times recorded. The critical times comprising the transfer consist of the time to get the data from the disk and transmit the data block (typically a 128-byte CP/M sector), and the time to store the block on the destination disk.

In the transmission from the server to the node, the transfer utility in the node, operating at the application layer in the ISO Model, starts the transfer by creating a new destination file on the local disk. The source file is requested to be opened, but this command is intercepted by the NDOS and redirected to the server over the network. The node then issues a series of "read next sector" commands which are also intercepted and passed along to the server. With each command, the server goes to the disk, reads the sector and transmits the CP/M logical sector to the node. While the server disk sector is being read, the node processor, having nothing else to do, can take the data at almost a full 9600 baud. Once the 128 bytes of data has been received, the file transfer utility stores it in memory, from which it is written to the disk upon a full memory condition or end of file.

In the transmission from the node to the server, the operation is the reverse. The file transfer utility reads the file (or as much of the file as will fit) into memory. The utility then issues a series of commands that contain the 128-byte sectors and request a "write next sector" operation. These commands are intercepted by the NDOS and sent to the server over the network. Since the server must support activities of other nodes and from the local consoles, it cannot receive the data at a full 9600 baud. The reason for this is that characters arriving at a serial I/O port at 9600 baud arrive at 100-microsecond intervals. With each character, an interrupt is generated to the server processor. The processor must identify the interrupting hardware, input the character, store it in the appropriate buffer, perform other housekeeping, then return to the interrupted process. This procedure consumes approximately 50 microseconds. It is easy to see that two nodes inputting data to the server could easily overrun the processor. To maintain reliable data transmission without losing data or having to retransmit many packets, a delay of approximately 8 milliseconds is inserted between each character that is transmitted. As a result, the best case transmission time for the "standard" 6-kbyte file from the server to the node is approximately 40 seconds and the node to server time is 85 seconds. The best case is when the server has no other node or local demands on it. With another node placing a similar demand on the server and with a local console actively listing a file, the server-to-node time goes from 40 up to 60 seconds and the node-to-server time goes from 85 to 95 seconds. These times indicate some elasticity of the server in being able to accept the additional loading. With the 8-millisecond-per-character delay decreased to 6 milliseconds, the best-case server-to-node time decreases from 40 to 35 seconds and the best-case node-to-server time decreases from 85 to 65 seconds. But when the load is added to the server, the communication breaks down due to lost messages. In an environment where there are up to 16 nodes making demands on the server, larger delay times for transmission to the server will be necessary.

V. Conclusions Based on Current Status

The current implementation state of the prototype local area network has shown that the use of CP/NET software provides an acceptable networking environment in all respects save one: speed of transmission time. The CP/NET software provides a standard, commercially available implementation of the ISO Model layers 3 through 6, providing full networking support for the application programs running on ISO Model layer seven. The customized implementations of layers 1 and 2 are solely dependent upon the available hardware for the physical layer and the implementations of the XIOS and SNIOS for

[2] Wordstar is a trademark of MicroPro International Corp., San Rafael, Calif.

the data link and network layers. The fact that the processor, particularly on the server but also on the nodes, spends so much time doing the byte-at-a-time I/O in these layers not only places a physical limitation on the network transfer speeds but also deprives application processes of valuable compute time.

To make a CP/NET implementation of a local area network a viable alternative, the low level communications tasks must be removed from the processor through the use of an intelligent or semi-intelligent communications front end.

VI. Future Directions

Communication with the nodes, which currently runs at an effective rate of less than 9600 baud, would be greatly enhanced by the addition of an intelligent data communications front end. The key problem is the fact that I/O to the network is performed through serial ports in a byte-at-a-time fashion. By performing the network I/O through a DMA operation to a board connected to Ethernet or some other high-speed medium such as broadband or fiberoptics, the communicating processes are relieved of much of the most time-consuming work. The processor, operating in the XIOS or SNIOS, no longer does the character I/O but instead simply indicates to the communications front end the memory location of the message. The communications front end takes the message through a DMA operation, sends it, receives the reply and places it in memory, only then notifying the processor that the reply has arrived. The processor during this time is free to perform other tasks. Operation of nodes with no local disks, previously unworkable due to the speed of the network, would now be more plausible.

The new version of MP/M, MP/M II provides additional features and capacities for the server node of the network. This version of the software supports up to 16 disk drives of up to 512 megabytes each, up to 16 printers, and up to 16-character I/O devices. Of the 16-character I/O devices, up to eight can be local consoles to MP/M and up to 15 can be CP/NET nodes. This version of MP/M provides file passwords, the ability to lock files and/or records within files for multiple user protection, automatic archive tagging for backing up files, and time/date stamping of files.

The development of software tools and managerial tools resident on the system will enhance the productivity of the users. The use of MP/M II, a new version of MP/M, allows for up to 400 kbytes of RAM, some of which could be used for a memory-resident data base manager and memory-resident virtual disk storage area. This arrangement would allow for the implementation of powerful data base management techniques.

New versions of CP/NET software will consist of ROM-able system modules that allow the operation of network nodes that have no disk drives at all. This software, coupled with the higher-capacity communication links, will allow bringing needed managerial and developmental tools to users that have only a terminal to upgrade and have no need for local disk storage.

Preliminary tests have indicated the plausibility of file transfer software that would interface the MODCOMP 7870 to the CP/NET server. This facility would provide needed offloading of file editing, archiving and software development version control. Interfacing of CP/NET to the VAX 11/780 could be accomplished with the possible procurement of commercially available software packages that meet the corresponding ISO Model layers of the CP/NET software. Hardware and software drivers already exist that meet ISO Model layers 1 and 2 of the Ethernet specification for the VAX 11/780, MODCOMP, multibus, and S-100 bus.

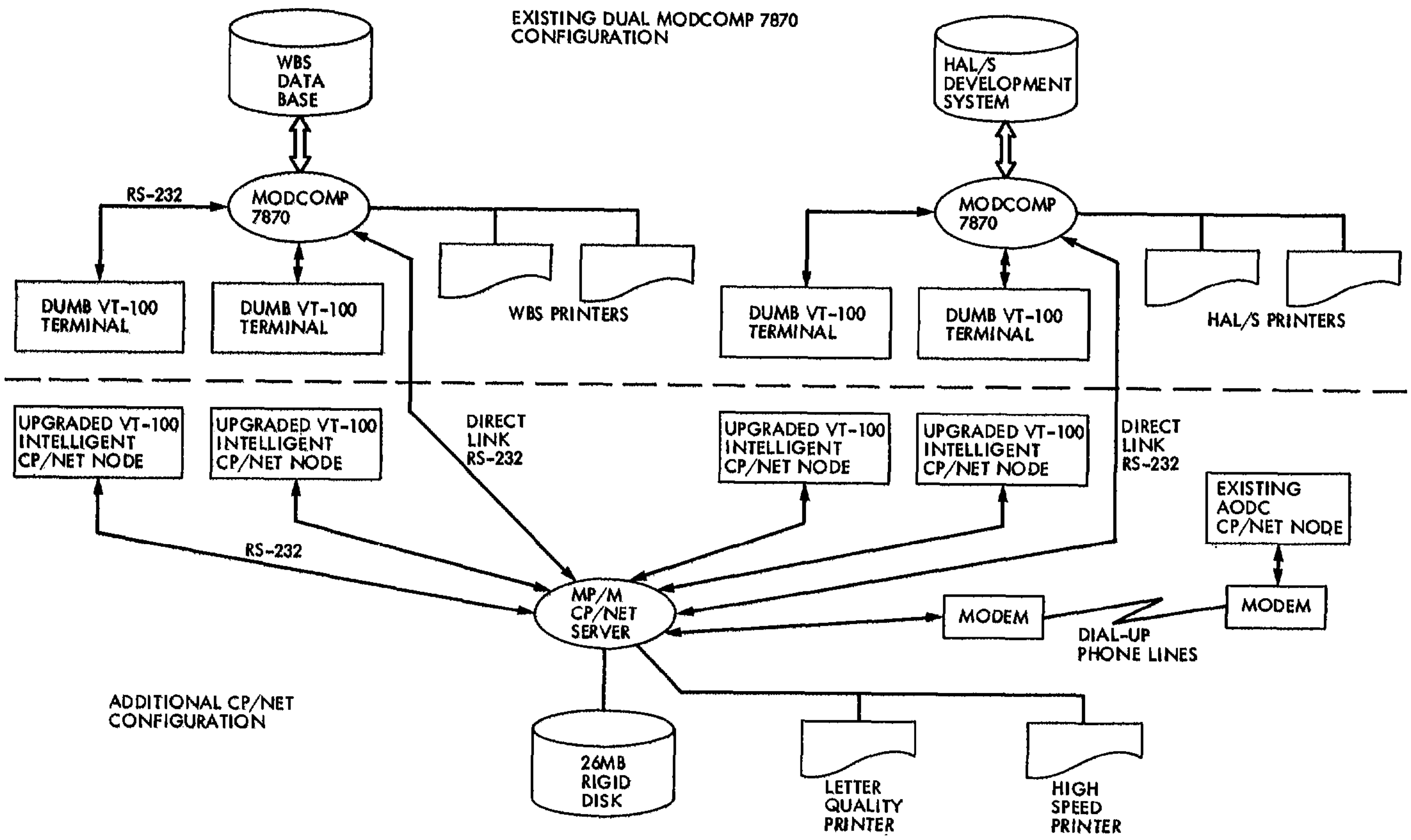

Fig. 1. MADNET initial upgrade configuration

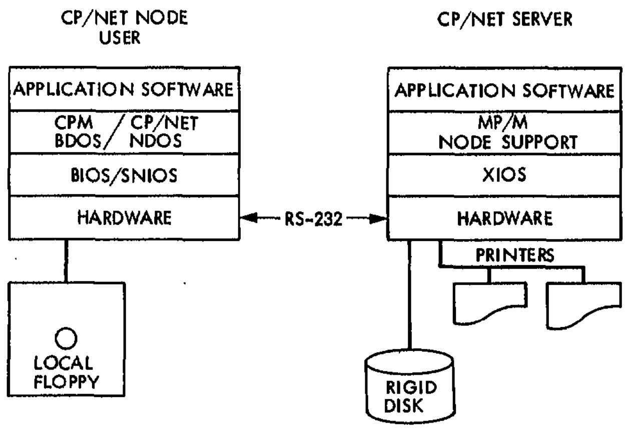

Fig. 2. Software/hardware layering

ISO MODEL LAYERING	CP/NET NODE	CP/NET SERVER	
7 - APPLICATION LAYER	- MANAGEMENT AND DEVELOPMENT TOOLS	DBMS SYSTEMS	
6 - PRESENTATION LAYER	- CP/NET NDOS	NODE SUPPORT	STANDARD CP/NET SOFTWARE
5 - SESSION LAYER	- CP/NET NDOS	NODE SUPPORT	
4 - TRANSPORT LAYER	- CP/NET NDOS	NODE SUPPORT	
3 - NETWORK LAYER	- CP/NET NDOS	NODE SUPPORT	
2 - DATA LINK LAYER	- SNIOS	XIOS	CUSTOMIZED NETWORK SOFTWARE AND HARDWARE
1 - PHYSICAL LAYER	- HARDWARE	HARDWARE	

Fig. 3. ISO Model mapping to CP/NET