DISTRIBUTED PROCESSING
IN HIGH VOLUME TRANSACTION PROCESSING SYSTEMS

A paper presented to the 10th International Meeting
of
The Hewlett-Packard General Systems Users Group at Orlando, Florida, U.S.A.
April 27 through May 1, 1981

By

Thomas L. Fraser
Systems Research Inc.
2400 Science Parkway
Okemos, Michigan
USA

Tuesday E-5 - 01

I. Introduction

Distributed processing concepts are frequently at the center of the implementation of transaction processing systems. The strengths of the HP3000 in these types of systems make it a desirable alternative from several standpoints. However, limitations of the HP3000 require that new approaches be used if successful high volume transaction processing systems are to be implemented.

One innovative solution applies the theory of distribution to the HP3000 itself. By vertically distributing the processing to be done on the HP3000, it becomes a viable solution to a much broader range of business problems. Systems Research Incorporated has effected this vertical distribution through the development of an HP1000-based front-end to the HP3000.

The SRI front-end dramatically expands the power of the HP3000 both in data communcations and data processing. Moreover, it makes new options available both in on-line linkages to other processors in distributed systems, and in other areas of data communications and terminals.

II. A Definition of Transaction Processing

A TRANSACTION is a phenomenon of the real world. It is a logical unit associated with an event or series of events in a system external to the computer. When one makes an airline or hotel reservation, a transaction occurs. Invoicing a customer, booking a suspect into jail, or inquiring on the status of inventory are all examples of transactions. Many transactions have nothing whatsoever to do with computers (e.g., buying the morning newspaper from a vendor on the street).

As can be seen from the examples, transaction processing is nothing new. Transactions have been occurring since the beginning of civilization, have been processed manually for thousands of years, and have been batched and processed on computers for decades. What is relatively new, is on-line transaction processing on mini-computers. When a system is on-line, transactions are processed individually as they occur in the real world. One can readily see that having an airline reservation system on-line is desirable, and there are many other on-line systems which will provide benefits that are not possible in batch systems. In the past it has been impossible or too costly to put most systems on-line. Today the improvements in hardware and software technology have changed this a great deal. Specifically, the HP3000 provides an environment which can be used to develop very high performance and effective on-line transaction processing systems.

III. The Transaction Monitor

A transaction, being an external occurrance, must be input to the computer system before it can be processed. In an on-line system this is usually accomplished through a terminal, frequently a CRT terminal. A transmission from a terminal to the host computer is defined a MESSAGE. A transmission to the computer and a response back to the terminal is a MESSAGE-PAIR. A transaction may involve one or multiple messages or message-pairs. Most transactions which are entered from CRT type terminals will involve multiple messages, and are frequently referred to as interactive.

There are several functions which are common to transactions in general, and do not depend on the specific application itself. Therefore, it is possible for these functions to be handled by software or firmware external to the application program. Examples of these functions are control and management of the network, data communications between the terminals and the host computer, logging of transactions, queueing of messages, routing messages, controlling access to the computer, handling screen formats, spooling of output print files, and providing facilities for restart and recovery in the event of a system failure. Some of these functions may be performed by the host operating system, but frequently these functions are grouped in a program (or programs) which is logically between the terminals and the application program, and is called a transaction monitor.

IV. Transaction Processing vs. Timesharing

In timesharing while user capabilities can be restricted, in principle users have broad latitude in what they can do, and the activities of the others on the system are transparent. The system is user and terminal oriented and provides an environment where each user can be simultaneously performing totally different tasks.

In transaction processing user capabilities are pre-defined and very limited. The system is transaction oriented and provides an environment where several users can perform the same or similar tasks simultaneously. Our earlier example of the airline reservation system illustrates; the reservation clerks are executing very similar transactions and are very restricted in capability (e.g., none of them can execute a compile or use Query).

The HP3000 uses MPE, a timesharing operating system. MPE associates a user with a stack. There is a one-to-one correspondence between users (terminals) and stacks. Each time another user logs on, another stack and another process provides for maximum flexibility and power for the users, but severely limits the number of users (terminals) because of the load that is placed on the resources of the HP3000.

V. Performance Limitations in the HP3000 Environment

Performance will vary from system to system depending on many factors such as what needs to be done, how well software is written, how well data structures are defined, etc. but some general measures are useful.

There are many ways to measure performance, but in a transaction processing environment it is the number of transactions which can be handled per unit time and the number of concurrent users which can be supported with acceptable response time.

The performance limitations of the HP3000 as measured by these two means are discussed in the "Hewlett-Packard Performance Guide"[1]. An approximation of two charts from that publication are included for convenience (Figs 1 and 2). Based upon these data, even a large Series 44 (with 4MB) reaches capacity at a transaction rate of 10,000/hour, and response time gets bad at about the 60 terminal level. For a 2MB Series III with MPE III, these numbers are about 4,000 transactions per hour and 30 terminals. High performance is defined for use here as that beyond what the native mode HP3000 and MPE can yield.

The resources which limit performance can be generally classified as CPU, memory, datacomm capacity and available disk accesses. (They are not independent of one another. They are considered separately here for simplicity as we are only illustrating a point, not presenting a study on performance). If we are to achieve higher performance, we must either add to these resources, or make better use of those already available. As will be seen, the vertical distribution of some functions to a front-end will accomplish both.

Footnotes:
[1] Hewlett-Packard Performance Guide, March 1981

Figure 1

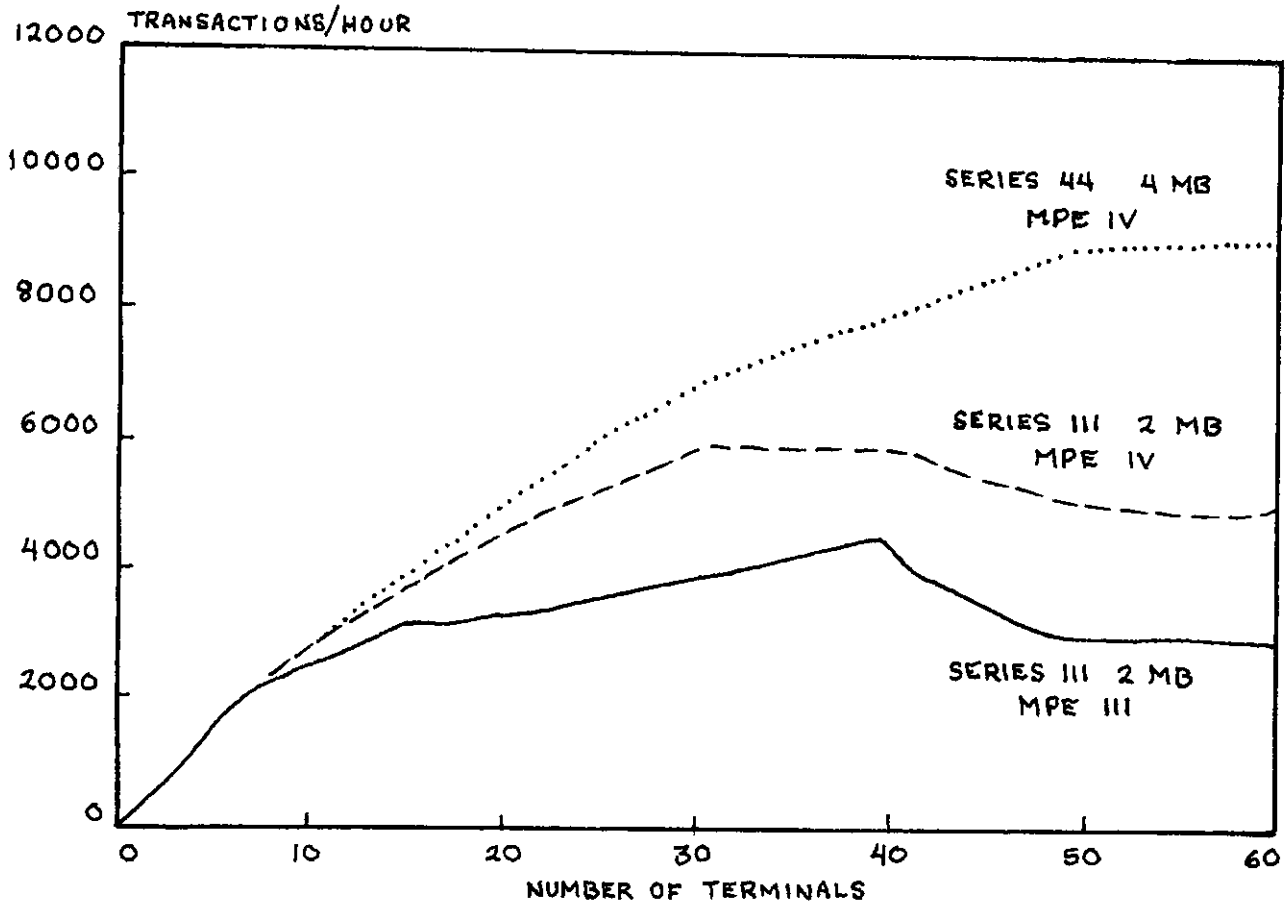
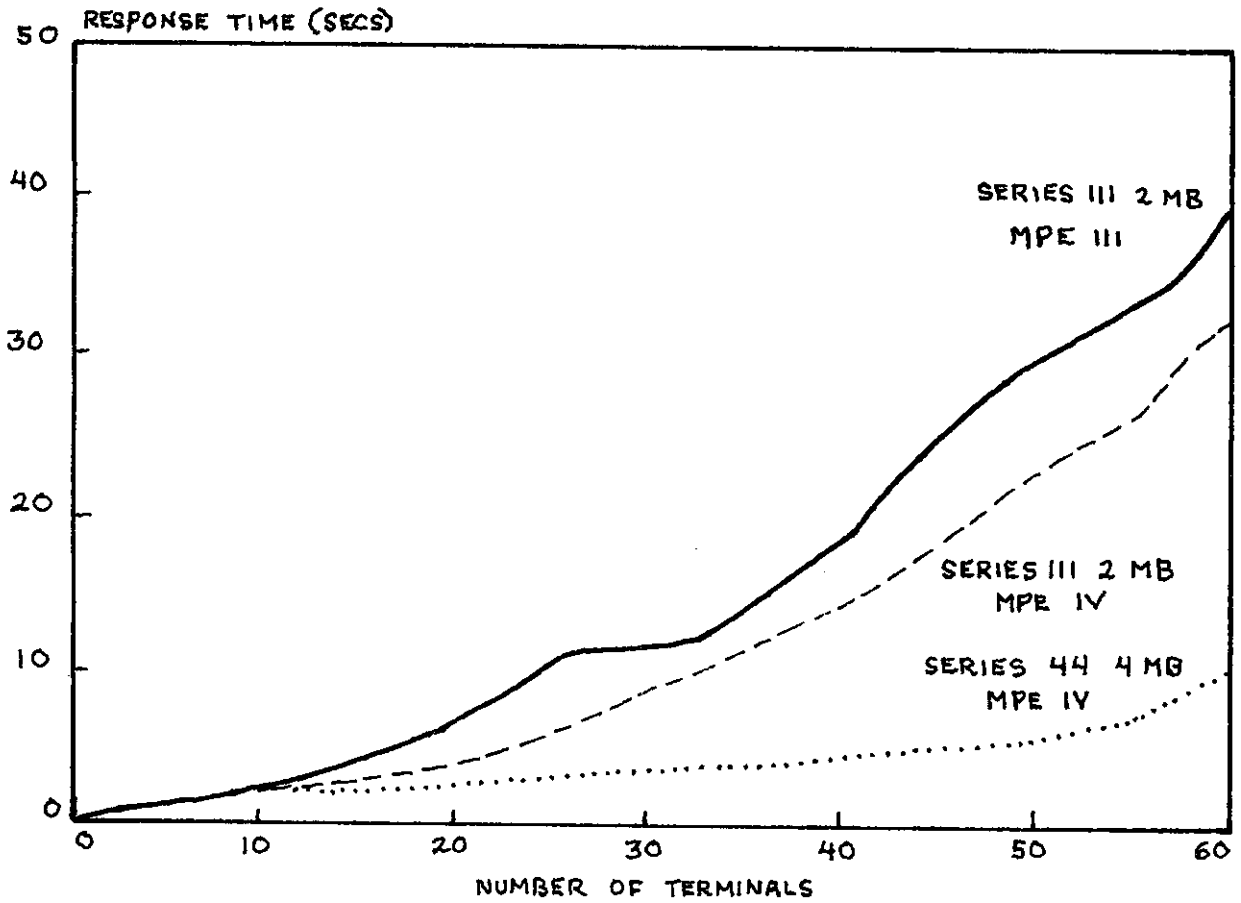


Figure 2



VI. Vertical Distribution Vehicle

The logical candidates for distribution to the front-end are of course those functions referred to earlier which are common to most all transactions. That is precisely what was undertaken.

The HP1000-E was chosen over alternatives because of reliability, price/performance, vendor quality, and support for user microcoding. All software and firmware was written specifically to perform the datacomm and transaction monitor functions. In effect, a specialized and dedicated operating system was written.

The HP1000 is interfaced to the HP3000 through the Universal Interface Card for Series III hosts and through HP-IB for Series 30, 33 and 44. Although the mechanics of the two interfaces differ, both achieve very fast and broad transfer of data between host and front-end with very little demand on host resources. An HP7906 disk is used by the front-end for large operating system tables and queue space, but because the HP1000 has 256KB (or more) of memory, queueing is usually done in memory. The datacomm line controllers complete the front-end hardware, and since they are also HP manufactured the entire system can be maintained by HP field personnel. Up to 240 high speed asynchronous lines or up to 70 high speed synchronous lines can be supported.

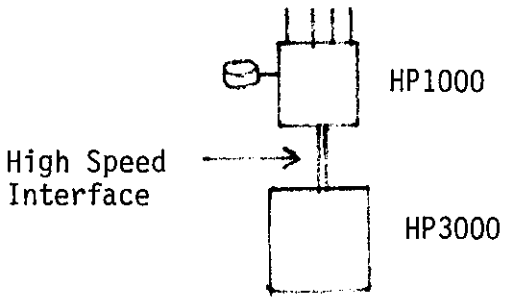


Figure 3: Basic Configuration

VII. Message Flow and Application Structure

When timesharing was discussed earlier, we saw that users and terminals were associated with processes and stacks. This is neither desirable nor necessary in a transaction processing environment. It impedes performance, control, and software development.

The front-end operating system associates transactions with processes. All executions of a particular transaction type can be routed to a single (or a small number of identical) process(es). The front-end treats all messages independently and transmits only data to the host. All forms, control characters and etc. are stripped off or inserted by the front-end. Since the application sees only data, it is as if there were only one terminal of unspecified type in the network.

A typical transaction will begin with a message from a terminal which begins with a slash, followed by a transaction mnemonic, followed by optional data. The front-end will check for transaction validity, as well as user and terminal authorization. If all checks-out, routing information is extracted from transaction tables, and the host process is checked for availability. If necessary the message is queued. When available, the message (data only) is sent to a switch program in the HP3000, which moves it to an extra data segment, and activates the application process. The user and station are logically linked to the process, and will be until the transaction is completed.

For detailed descriptions of structure, refer to the MCS3000 programming manual[2].

VIII. Features and Performance

The division of labor and specialization of resources, together with the addition of resources resulting from the vertical distribution, and the adoption of a transaction oriented architecture greatly expand the capacities and capabilities of the HP3000. Moreover, flexibility is enhanced and many otherwise unavailable features are brought to the HP3000.

Footnotes:

2 MCS3000 Programming Manual, Systems Research Incorporated, June 1980

As measured earlier, performance is several times what an HP3000 alone can do. Transaction rates of up to 40,000 per hour have been achieved on Series IIIs with MPE III, and although no Series 44s have been installed with front-ends at this writing, even higher rates are likely then. The maximum number of terminals currently installed on a single front-end is about 600 (see Appendix 1), but many times that could be supported.

Other performance limitations are 250,000 bits per second throughput, up to four simultaneous hosts (HP3000's or Burroughs Medium Systems) and up to 500,000 characters per second transfer between host and front-end.

Front-End Resident Messages Control System - The front-end message control system offers: Transaction-based message routing and/or predefined message linkage support; application data-save areas; a five-level user and terminal security system; front-end forms files, with automatic data-fill; application program control; and dynamic network management. With the front-end host resources are directed toward the support of user applications, not data communications or message control functions.

Protocol Flexibility - Supports most standard data communication protocols; also, SRI can develop specialized protocols tailored to specific user requirements. This flexibility allows price and capability to become the key criteria in selection of terminals and remote devices. (See Appendix 2.)

Application Independence - The front-end totally insulates application programs from data communication and message control functions. The front-end also provides a prototype application handler for simplified development of new application programs. Together, these yield efficiencies in application program development and maintenance.

Device Independence - The front-end control character mapping and automatic forms-fill features allow device-independence at the application program level. Specific application programs can be developed without regard to the characteristics of a given terminal. Device-independence reduces application program development time; in addition, changes in terminal hardware no longer obsolete application programs.

Dynamic Network Configuration and Maintenance - With the front-end, the network is defined using on-line transactions. Most network parameter changes can be made dynamically; network down-time thus is eliminated for recompilation of front-end or host application programs. Refer to MCS3000 Reference Manual[3].

Other Features - The front end provides: Audit and recovery; on-line forms generation and maintenance; remote print spooling; and on-line access to summary network statistics.

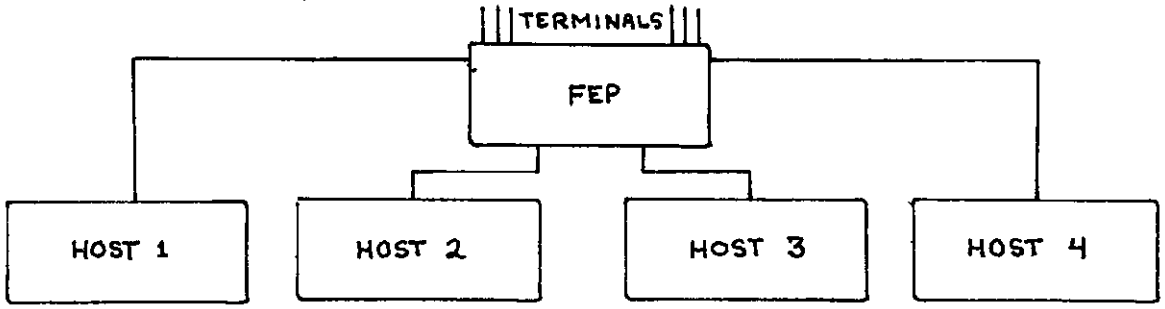
Multiple Configurations for On-Line Processing

The basic front-end is configured with a single front-end processor (FEP) interfaced to a single host computer; alternative configurations include:

- Simultaneous interfaces to as many as four hosts.
- Multiple front-end processors interfaced to a single host.
- Redundant front-end processor configurations for single- or multi-hosts.

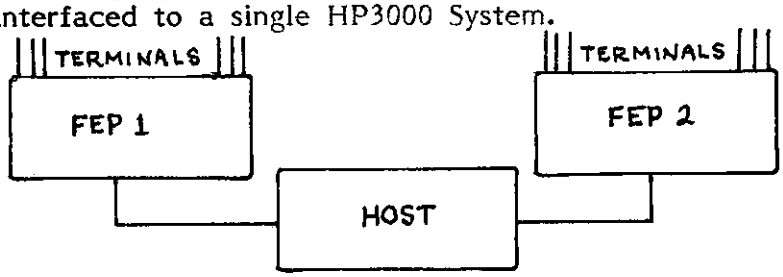
Single FEP to Multiple Host Configurations

The front-end provides an integrated front-end data communication and message control system for the multiple host environment. Any combination of HP3000 Systems can be interfaced to a single front-end. Alternatively, one or more Burroughs Medium systems can be used in conjunction with the HP3000 host(s).

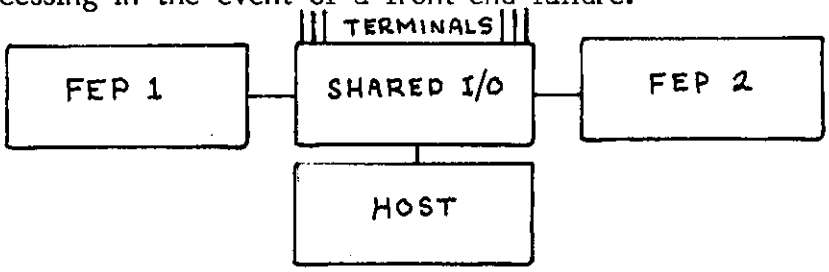


Footnotes:
[3] MCS3000 Reference Manual, Systems Research Incorporated, June 1980

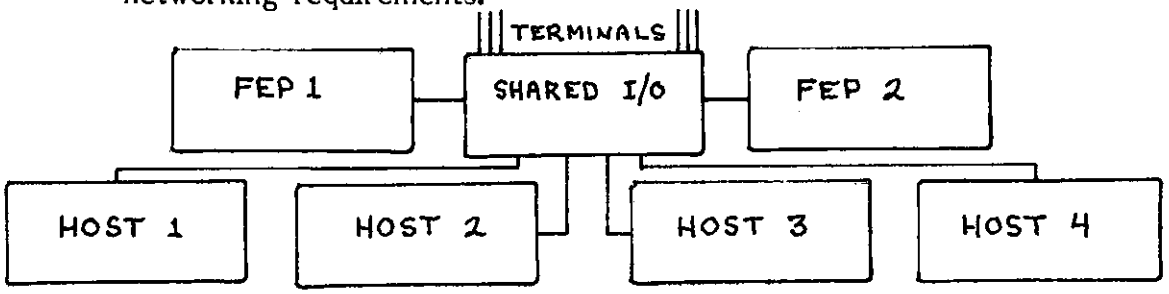
Multiple FEPs to Single Host Configuration - Where network requirements dictate discrete front-end processors, multiple FEPs (limited only by the number of HP host interface slots available) can be interfaced to a single HP3000 System.



Redundant FEPs to Single Host Configuration - Redundant front-end processors, sharing all I/O interfaces (SHARED I/O) to the terminal network and the host, provide automatically switched front-end processing in the event of a front-end failure.



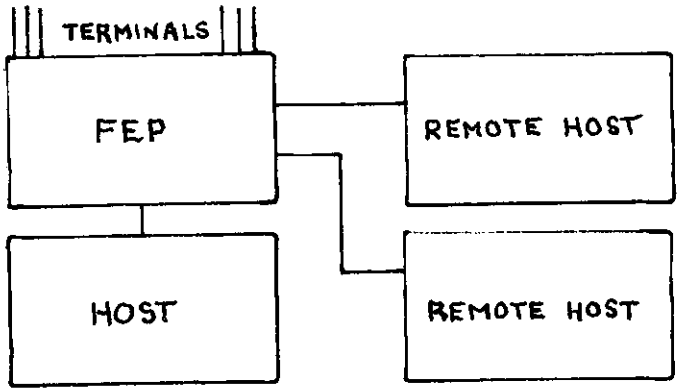
Redundant FEPs to Multiple Host Configuration - Combining the unique front-end features of redundancy and multiple host support provides a configuration suitable for the most demanding of networking requirements.



The front-end has been designed to provide maximum flexibility. It supports the above configurations, but alternative configurations are available to meet specific networking requirements. For example:

Store and Forward - With a data communication link, the front-end can store and forward messages to remote processors. These messages can originate from: 1) Terminals interfaced to MCS3000 Model 200; 2) host resident application programs; or 3) other remote processors.

By utilizing "inverse protocols", the front-end appears to the remote host processor as a terminal or terminal controller (e.g., IBM 327X, IBM 2780, Burroughs TD830, etc.). The store-and-forward feature enables an HP3000 user to interface with a remote (IBM, Burroughs, etc.) host network.



Remote Diagnostics and Maintenance - To enhance SRI technical support the front-end can provide remote diagnostics and maintenance which enables SRI technical support personnel to monitor, isolate and correct front-end code, memory and disk files remotely, without halting on-line operations.

IX. Summary

The SRI front-end, MCS3000, significantly enhances the HP3000. The capacity of the HP3000 to do transaction processing is greatly increased because of the distributing of data communications and transaction monitor functions to a specialized processor. Moreover, many options are opened to the implementers of systems, such as the multiple host configurations, redundant configurations, and the ability to interface foreign terminals and processors; all new options for distributed processing with HP3000's.

Appendix I

In August of 1978 a major brokerage firm headquartered in the midwest began a process to replace its aging communications system. The company was then communicating with its 200 branch offices via low speed teletypes. During the inquiry, the firm came to realize that they might do much more than just replace their TTYs. Thus in January of 1979 they selected Systems Research of Okemos Michigan to install an online turnkey computer system and terminal network which was to handle both communications and data processing functions.

SRI, working closely with the brokerage firm's staff, began immediately to design a comprehensive system centered around MCS3000, SRI's HP1000 and HP3000 based communications and transaction processing product. The resulting plan called for a phased implementation which would have as its first goal the replacement of the existing "network". Installation of the Hewlett-Packard hardware began in April of 1979.

By the fall of 1979 all of the old teletype equipment and slow asynchronous lines had been replaced by CRT and printing terminals multidropped on 2400 baud synchronous lines. By the end of 1980 due to the growth of the firms business, there were over 250 offices being supported, and the network consisted of more than 600 terminals. Additionally, an on-line datacomm link to a non-HP remote mainframe is supported for accessing a highly volatile and time-critical data base. The entire network is controlled and supported from a single HP1000 minicomputer with SRI's MCS3000 software. (See figure A).

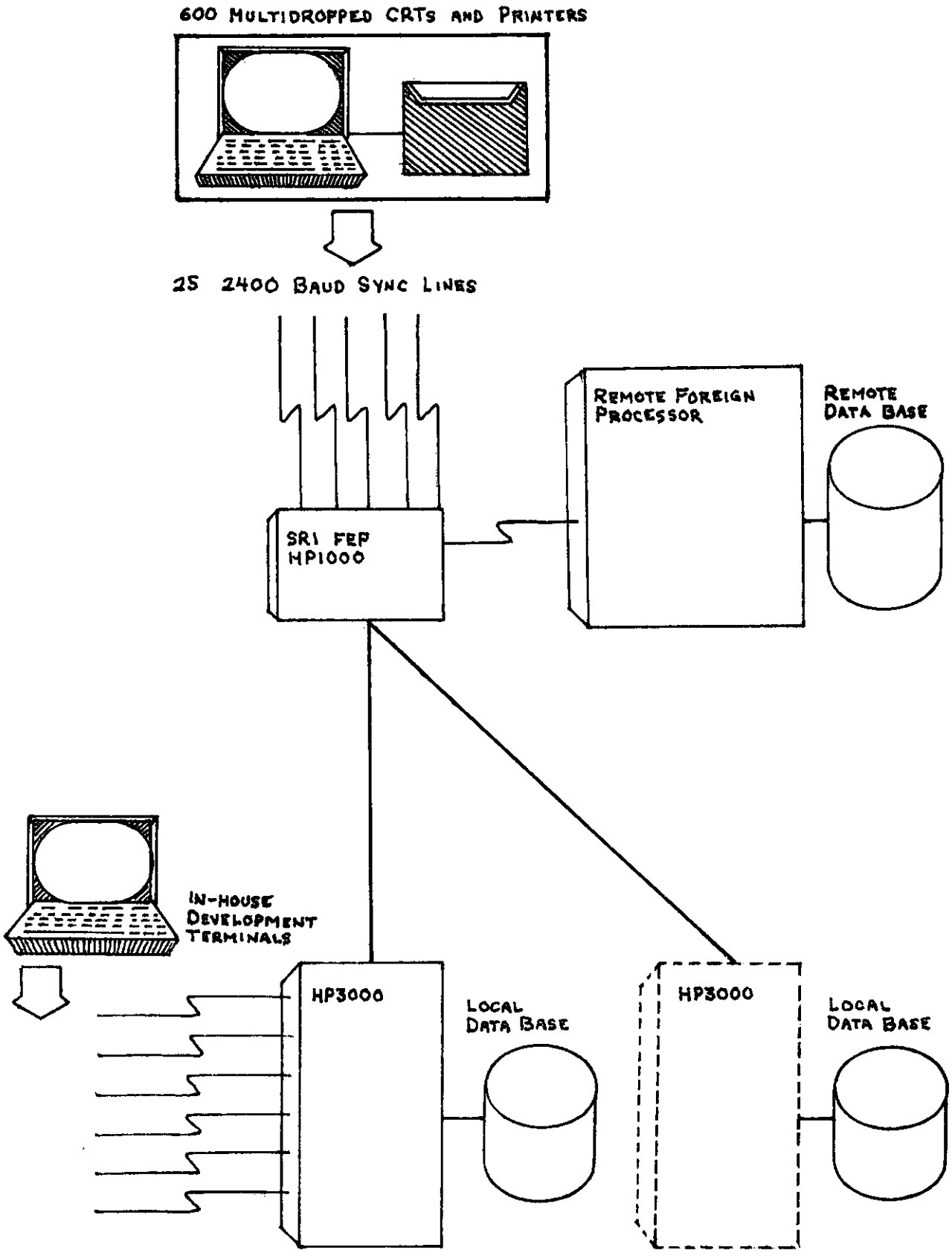
The HP1000 is linked through high-speed (400,000 char/sec) HP interfaces to an HP3000 Series III which handles the data processing functions. This vertical distribution provides, in effect, a division of labor which makes possible much higher throughput on the HP3000, as well as the obvious expanded data communications. The HP1000 is configured with 256KB of memory and 20MB of disk; the HP3000 has 1.5MB of memory and 375MB of disk, as well as a pair of tape drives and a high speed line printer.

Appendix 1 Cont.

There are four major applications currently being handled by the system, trades, quotes, research, and branch communications and administration. Each of the major systems has a single program in the HP3000 which handles all transactions for that application. The message volume being processed by the current configuration exceeds 80,000 daily and has reached a level of 15 per second on peak trading days.

Current development plans include the addition of another HP3000 to the configuration in 1981 to accommodate between 5 and 10 additional applications. At least three of these applications will require on-line/realtime computer to computer links with various securities service organizations. These links will be effected through the HP1000 front-end. Additionally the number of offices supported is expected to reach the 350 mark in 1981 demanding a minimum of 750 terminals to be supported by the configuration.

Appendix I, Figure A.



Appendix II.

SRI PROTOCOLS

Name	Description
PTTY	Character Mode Teletype
P264X	Block Mode HP 264X
PBIPP	IBM Bisync Point-to-Point (2780/3780)
PBIMP	IBM Bisync Multi-Point
PBPS	Burroughs Poll/Select
PADM	Lear-Siegler ADM-2
PBRJE	Burroughs Remote Job Entry (761)
PBPPCT	Burroughs Point-to-Point/Contention
PBPPBT	Burroughs Point-to-Point/Batch
PBPPCV	Burroughs Point-to-Point/Conversation
PNCR270	NCR 270 Basic Variant
PTTEL	Bell 407C Transaction Telephone
PAZUR	Azuredata Scorepad Terminal
PTIC	TI 742 Cassette Station
PVSET	VUSET device at Pan Am Bank
PNCR270A	NCR 270 Variant A
PNCR270B	NCR 270 Variant B
PNCR796	NCR 796 Basic Variant
PNCR796A	NCR 796 Variant A
PIDEN	IDENTICON Bar Code Reader
PIBM2260	IBM 2260
PVTRXI	Votrax In
PVTRXO	Votrax Out
PCARD	DATA CARD (IBM 2740 Subset)
PBNCR	Bisync Transparent N.C.R. (3270 var)
PCLETS	California L.E.T.S. (2780.3780 var)