

MULTIPLE COMPUTER NETWORKS AND INTERCOMPUTER COMMUNICATION

Lawrence G. Roberts
Advanced Research Projects Agency
Washington, D. C.

There are many reasons for establishing a network which allows many computers to communicate with each other to interchange and execute programs or data. The definition of a network within this paper will always be that of a network between computers, not including the network of typewriter consoles surrounding each computer. Attempts at computer networks have been made in the past; however, the usual motivation has been either load sharing or interpersonal message handling. Three other more important reasons for computer networks exist, at least with respect to scientific computer applications. Definitions of these reasons for a computer network follow.

Load Sharing: Both the program and data are transmitted to a remote computer to equalize the load between the two facilities. This type of operation requires identical computers or languages. When a given machine is loaded, consideration can be given to processing the program on another machine. Many determinations must be made before an alternate machine is used (is there an alternate machine for which appropriate software exists, is that alternate machine in a condition to handle the program, will more time and dollars be spent on transmission than if the user waits until the original machine is available, etc.). Such determinations are very difficult and the gain only moderate, hence load sharing is not a major consideration here. However, it is felt that some load equalization will occur in any computer network.

Message Service: In addition to computational network activities, a network can be used to handle interpersonal message transmissions. This type of service can also be used for educational services and conference activities. However, it is not an important motivation for a network of scientific computers.

Data Sharing: The program is sent to a remote computer where a large data base exists. This type of operation will be particularly useful where data files are too large to be duplicated economically. Frequently geographically dispersed individuals need to access a common data base. Access to this data base may be required simply to make an inquiry or may involve

executing a complex program using the data base. Use of a single data bank will save hardware required to store the information and will eliminate the need for maintaining multiple files. The term "single data bank" does not necessarily mean the storing of only one copy of each basic data file. This type of use is particularly important to the military for command and control, information retrieval, logistics and war gaming applications. In these cases, one command would send a program to be executed at another center where the data base existed.

Program Sharing: Data is sent to a program located at a remote computer and the answer is returned. Software of particular efficiency or capability exists on certain machines. For example, if machine Y has a good LIST processor, it may be more efficient for users whose local machine is X to use Y for LIST processing jobs. Even if a LIST processor exists for X, the time to execute the program on Y may be sufficiently less than the time to execute on X that the total time (and/or cost, including transmission) may be less. The use of specialized programs at remote facilities makes possible large gains in performance. Perhaps even more important is the potential saving in reprogramming effort.

Remote Service: Just a query need be sent if both the program and the data exist at a remote location. This will probably be the most common mode of operation until communication costs come down. There will be a tendency for other cases to migrate toward this type of operation. For example, in a graphics application, the program would be available or created on the remote computer and it would generate the data structure in its own computer. It would modify and update the data structure from network commands transmitting back display changes. This category includes most of the advantages of program and data sharing but requires less information to be transmitted between the computers.

The advantages which can be obtained when computers are interconnected in a network such that remote running of programs is possible, include advantages due to specialized hardware and software at particular nodes as well as increased scientific communication.



Specialized Hardware

It is felt that new machine configurations can provide improvement factors of from 10 to 100 in the problem area for which they were designed. In some cases very large core and disk will substantially improve performance on existing machines. In other cases the improvements will be brought about by introduction of new systems such as ILLIAC IV¹ and macromodular machines.² A network is needed to make full use of machines with specialized efficiency and with a network the development of such computers will be enhanced.

Specialized Systems Software

Handling jobs of widely varying sizes, particularly when initiated from many locations, presents an extremely difficult scheduling problem for any single machine. A large machine serving a number of smaller machines may provide significant improvements in efficiency by alleviating the scheduling problem. Small time sharing computers may be found to be efficiently utilized when employed as communication equipment for relaying user requests to some larger remote machine on which substantive work is done. What is envisioned is a system in which the local machine serves some limited needs of the user while substantial requirements are satisfied by a remote computer particularly well adapted to handling the problem.

Scientific Communication

Once it is practical to utilize programs at remote locations, programmers will consider investigating what exists elsewhere. The savings possible from non-duplication of effort are enormous. A network would foster the "community" use of computers. Cooperative programming would be stimulated, and in particular fields or disciplines it will be possible to achieve a "critical mass" of talent by allowing geographically separated people to work effectively in interaction with a system.

Approach

Basic Operation. The minimum requirement a system must meet to be eligible for membership in the network is a time-sharing monitor which allows user programs to communicate with at least two terminals. If this requirement is uniformly met, the network can be implemented without major change to the monitor at any installation, by the simple expedient of letting each computer in the network look conceptually upon all the others as though they were its own remote user terminals.

Figuratively speaking, we may think of the

computer-to-computer link in such a network as being the result of removing a user terminal from its cable on computer A, removing a user terminal from its cable on computer B, and splicing the two computer cableends together. Such a network might operate as follows: (See Figure 1) The user dials up his home computer, CA, from a console. He logs in normally by transmitting characters from his console to the monitor. He sets up a user program and this program, through the second channel, calls the remote computer, logs in, sets up the desired user program on the remote computer, submits parameter data to it and receives the results. Note that neither system was required to behave in an unusual fashion. The monitors did what they always do. The only requirement, as stated earlier, was that the user program be allowed to communicate with two terminals, its own user terminal and the remote computer. Most present-day monitors provide for such a capability.

A computer-computer network link as described above was established in 1966 experimentally between MIT Lincoln Lab's TX-2 computer and System Development Corporation's Q-32.³ Both nodes are general purpose, time-shared computers. This link allows programs on either computer to utilize programs such as compilers and graphics systems which exist only at the other node. The basic motivation was to test an initial network protocol, determine how well automatic dial up communications service worked, and determine the extend of the time-sharing monitor changes necessary. This has been done and the link is now utilized by users to increase their capability, thus providing more evaluation data.

Interface Message Processor. One way to make the implementation of a network between a set of time-shared computers more straightforward and unified is to build a message switching network with digital interfaces at each node. This would imply that a small computer, an interface message processor (IMP), would be located with each main computer to handle a communications interface. It would perform the functions of dial up, error checking, retransmission, routing and verification. Thus the set of IMP's, plus the telephone lines and data sets would constitute a message switching network (See Figure 2).

The major advantage of this plan is that a unified, straightforward design of the network can be made and implemented without undue consideration of the main computer's buffer space, interpret speed and other machine requirements. The interface to each computer would be a much simpler digital connection with an additional flexibility provided by programming the IMP. The network section of the IMP's program would be

completely standard and provide guaranteed buffer space and uniform characteristics, thus the entire planning job is substantially simplified. The data sets and transmission lines utilized between the IMP's would most likely be standardized upon, but as changes occurred in the communication tariffs or data rates, it would be more straightforward just to modify the program for the IMP's rather than twenty different computers. As soon as the need became apparent, additional small computers could be located at strategic connection points within the network to concentrate messages over cross-country lines. Finally, the modifications required to currently operating systems would be substantially less utilizing these small computers since there would be no requirement to find buffer spaces, hold messages for retransmission, verify reception of messages and dial up telephone lines.

ARPA Net

ARPA supports a number of computer research groups throughout the country most of which have their own time-shared computer facility. These researchers have agreed to accept a single network protocol so that they may all participate in an experimental network. The communication protocol is currently being developed. It will conform to ASCII conventions as a basic format and include provisions for specifying the origin, destination, routing, block size, and sum check of a message. Messages will be character strings or binary blocks but the communication protocol does not specify the internal form of such blocks. It is expected that these conventions will be distributed in final form during September 1967.

Figure 3 shows a tentative layout of the network nodes and communication paths. However, since most of the communications will be dial-up, the paths are just hypothetical. It is hoped that concentration and store and forward capability will be available through the use of Interface Message Processors. The development of the IMP's and the use of them at each node will allow store and forward operation as well as speeding the realization of a unified network.

There are 35 computers shown in Figure 3 at 16 locations, there being several computers at most locations. A rough estimate would place the number of consoles attached to the 35 computers by the end of 1967 at 1500 and the number of displays at 150. Assuming four characters per second for typewriters and 20 characters per second for scopes, the total I/O rate to the computers is 9000 char/sec. Estimating that 10% of this I/O communication rate will be forwarded to another computer in the network leads us to an average transmission rate of 60 char/sec per

location. Thus, given console type activity on the network (messages of from 10 to 1000 characters) the normal 2000 bits/second type communication should be sufficient at first.

Communication Needs

The common carriers currently provide 2 or 4 wire, 2 kc lines between two points either dialed or leased, as well as higher band width leased lines and lower band width teletype service. Considering the 2 kc offering, since it is the best dial up service, the use of 2 wire service appears to be very inefficient for the type of traffic predicted for the network. In the Lincoln-SDC experimental link the average message length appears to be 20 characters. Each message must be acknowledged so that the originator may retransmit or free the buffer. Thus the line must be reversed so often that the reversal time will effectively half the transmission rate. Therefore, full duplex, four-wire service is more economic and simpler to use.

Current automatic dialing equipment requires about 20 seconds to obtain a connection and a similar time to disconnect. Thus, the response time is much too long assuming a call is made only after a message arrives and that the line is disconnected if no other messages arrive soon. It has proven necessary to hold a line which is being used intermittently to obtain the one-tenth to one second response time required for interactive work. This is very wasteful of the line and unless faster dial up times become available, message switching and concentration will be very important to network participants.

BIBLIOGRAPHY

1. Slotnick, D. L., "Achieving Large Computing Capabilities Through An Array Computer," presented at SJCC (April 1967).
2. Clark, W. A., "Macromodular Computer Systems," Proc. SJCC (April 1967).
3. Marill, T. and Roberts, L. G., "Toward a Cooperative Network of Time-Shared Computers," Proc. FJCC (1966).

Figure 1

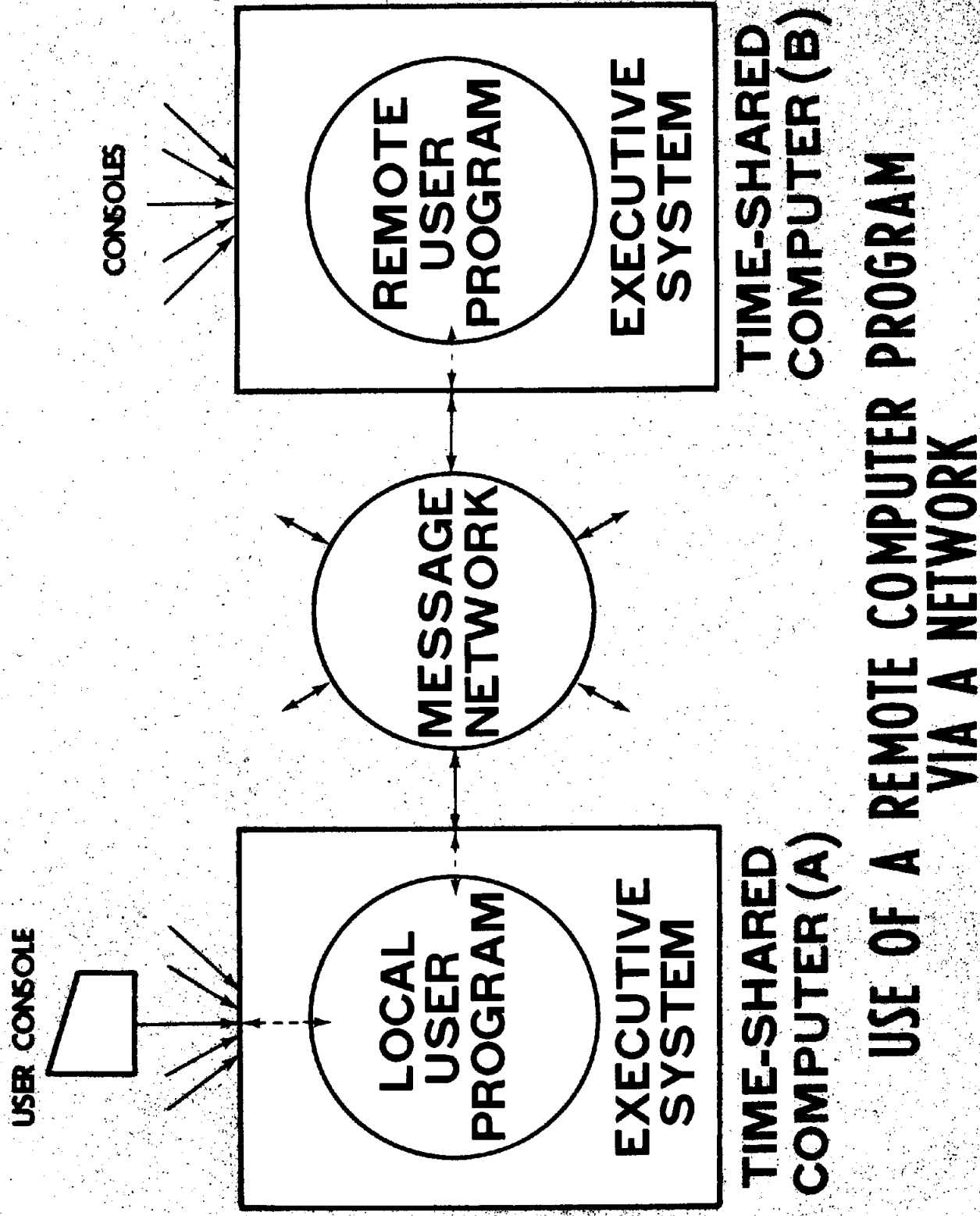
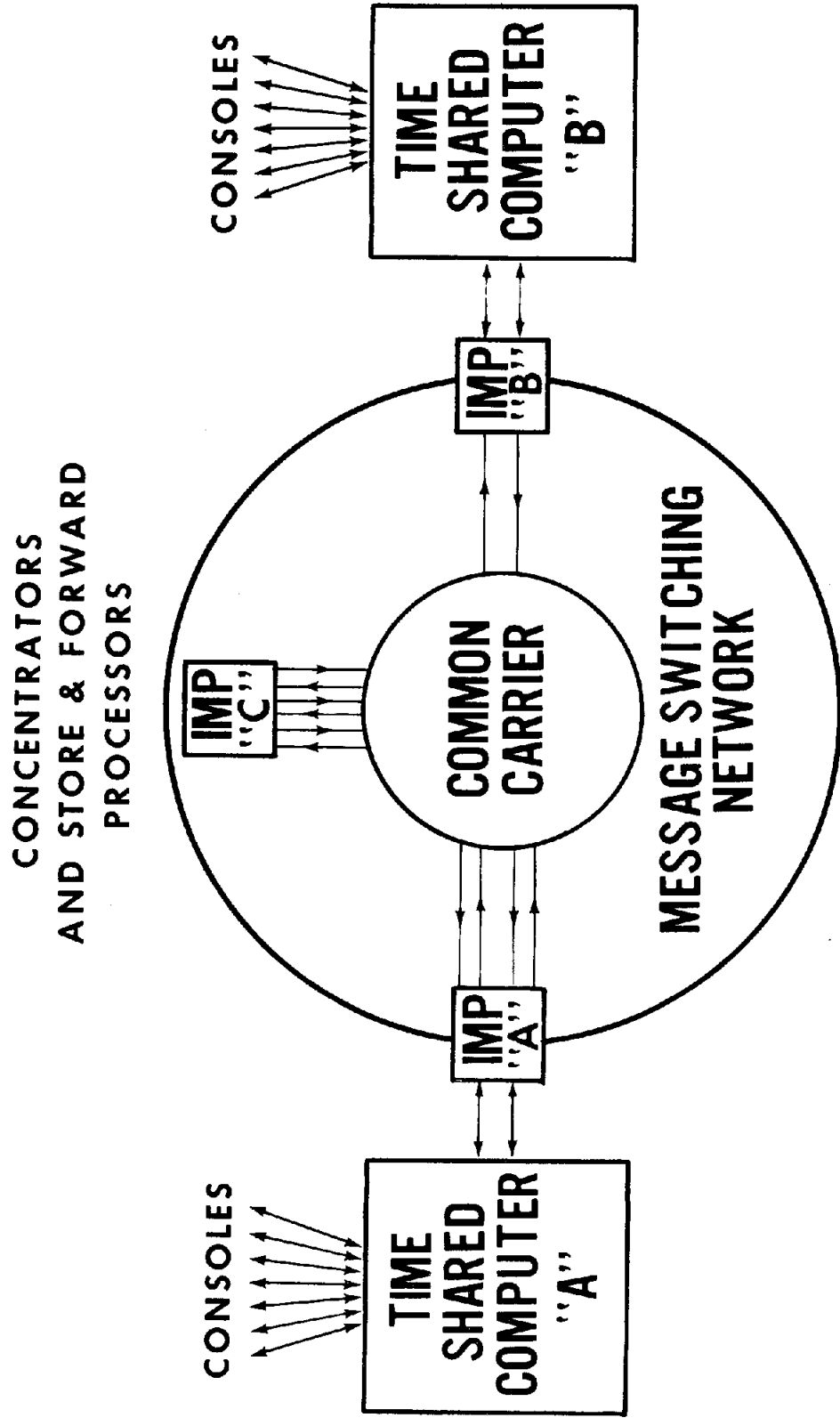
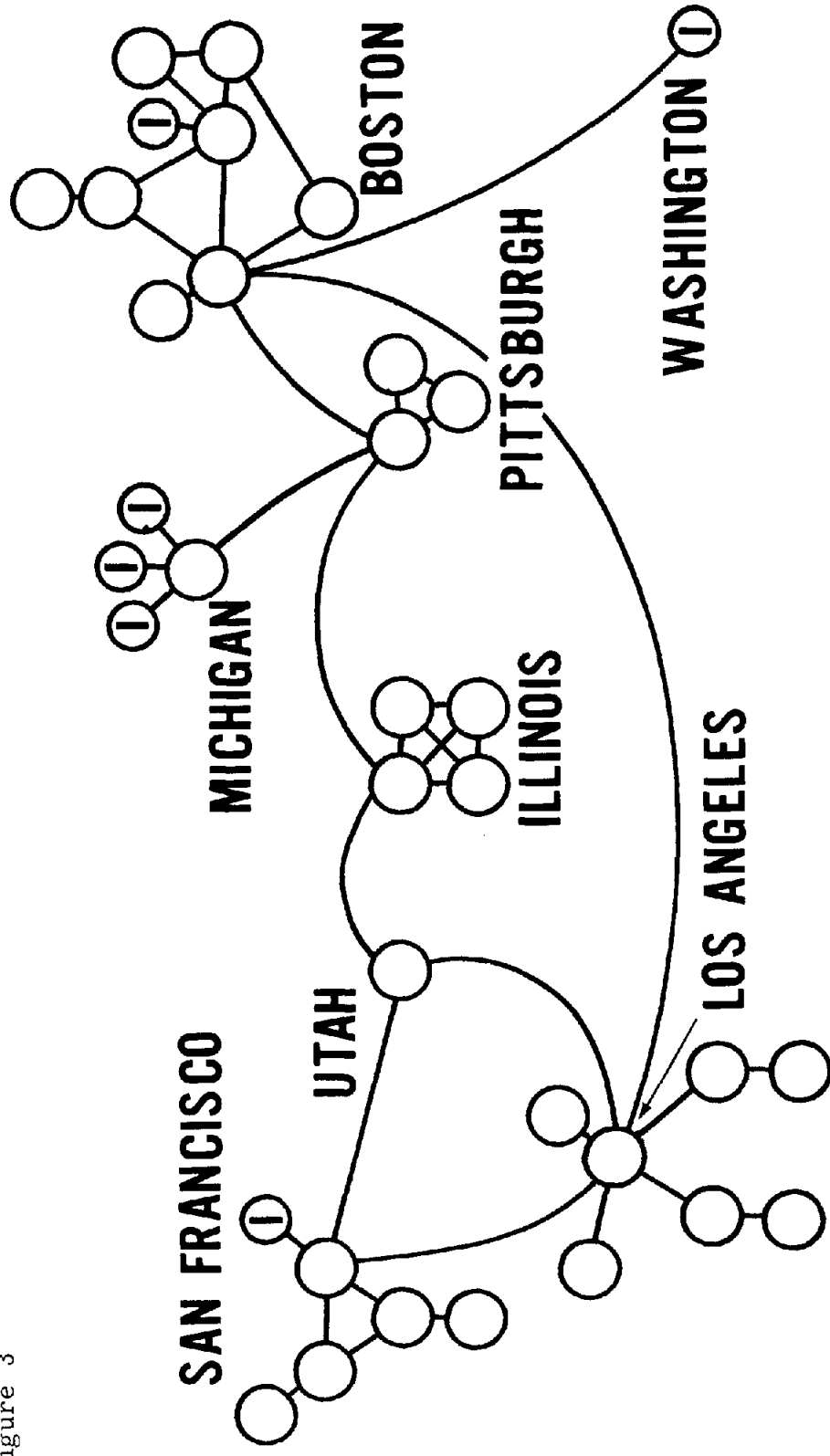


Figure 2



**USE OF INTERFACE MESSAGE PROCESSORS (IMP)
FOR ROUTING AND CHECKING INTER COMPUTER COMMUNICATION**

Figure 3



ARPA COMPUTER NETWORK*

○ REPRESENTS A TIME-SHARED COMPUTER

① REPRESENTS A SINGLE USER CONSOLE COMPUTER

* NOT FINAL, ESTIMATE AS OF JUNE 1967