Set III

BOLT BERANEK AND NEWMAN INC

CONSULTING • DEVELOPMENT • RESEARCH

BBN 2353

INTERFACE MESSAGE PROCESSORS FOR THE ARPA COMPUTER NETWORK

QUARTERLY TECHNICAL REPORT NO. 13
1 January 1972 to 30 April 1972

Submitted to:

Advanced Research Projects Agency
Washington, D.C. 22209
Attn: Dr. L.G. Roberts

This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract No. DAHC-15-69-C-0179.

Report No. 2353 Bolt Beranek and Newman Inc.

INTERFACE MESSAGE PROCESSORS FOR THE ARPA COMPUTER NETWORK

QUARTERLY TECHNICAL REPORT NO. 13
1 January 1972 to 30 April 1972

Submitted to:

Advanced Research Projects Agency
Washington, D.C. 22209
Attn: Dr. L.G. Roberts

This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract No. DAHC-15-69-C-0179.

TABLE OF CONTENTS

1. OVERVIEW

This Quarterly Technical Report, Number 13, describes aspects of our work on the ARPA Computer Network during the first quarter of 1972.

During this quarter three new IMPs were installed and two IMPs which had been previously installed were relocated. The 316 IMP originally installed at ETAC was moved to McClellan Air Force Base (Sacramento, Calif.) and the 516 IMP originally installed at Paoli was moved to NASA/Ames in preparation for the eventual attachment of the ILLIAC IV computer complex. A 316 IMP was installed at Tinker Air Force Base (Oklahoma City, Oklahoma) and Terminal IMPs were installed at ETAC and at USC (University of Southern California, Los Angeles). Thus, by the end of the quarter, the network contained 23 operational nodes, plus the BBN prototype TIP.

The TIP installed at ETAC during the first quarter was the first machine delivered with the magnetic tape option, which is described in Section 2.

In conjunction with the installation of IMPs at Tinker and McClellan, we delivered two special Host interfaces designed and fabricated at BBN for the Univac 418 III Hosts at those sites. Field testing of these special Host interfaces began late in the first quarter and the interfaces are expected to be fully operational early in the second quarter.

Late in the first quarter the BBN IMP, the prototype TIP, and the Network Control Center were moved to a new location within BBN. The move was accomplished in one day, a Saturday, with minimal disruption of normal network operation. Shortly

after the move a fourth Host (a PDP-1) was interfaced to the BBN IMP; this is the first instance of an IMP with four Hosts. This Host will be primarily used on the network as an adjunct to the Network Control Center and will not be available as a network resource.

The installation of a second IMP at NASA/Ames has provided a convenient opportunity for experimental use of high-bandwidth inter-IMP communication. Accordingly, a 230.4 kilobit/second line was installed between the two Ames IMPs during the first quarter. The operation of this circuit has proved satisfactory, and has failed to disclose any IMP hardware or software difficulties.

During the first quarter we have been actively involved in discussions regarding the possible extension of the network to Hawaii and other overseas points via earth satellite communications links, as well as investigating the use of long high-speed lines in the network. An important aspect of these modes of expansion is the requirement for an understanding of the IMP buffering needed to fully utilize communications links of these types. Accordingly, during the first quarter we studied the relationships among buffer requirements, line speeds, and line lengths under a variety of assumptions regarding packet size and acknowledgment strategy. The results of this study are presented in Section 3.

We continued our improvements to the Network Control Center during the first quarter. In addition to general efforts to improve operator procedures, one major project undertaken was the semi-automation of the processing of Host throughput and line throughput data. Previously, NCC-Teletype typescripts of 24 hour

summary data were processed completely by hand. Our new procedure is to punch this data on paper tape (as it is being typed) and process the paper tapes off-line. While this is far from an ideal solution, it does insure a higher degree of accuracy, as well as permitting multiple analyses of a month's data with no additional manual effort. In addition, we have documented the growth and current operation of the NCC in a paper (*The Network Control Center for the ARPA Network*) submitted to the 1972 International Conference on Computer Communication.

During the first quarter of 1972 we continued our studies of the proposed High Speed Modular IMP design and of the connection of a "remote batch" terminal to the TIP's MLC; some progress has been made in both of these areas. In addition, we have continued our involvement in the Network Working Group, particularly in the areas of periodic Host availability reporting and protocol development and refinement. During the first quarter we have been heavily involved in the development of "Remote Job Entry protocol" and also produced a revised version of the "Host/Host protocol" documentation. Also, we have continued to improve the capabilities of the Terminal IMP, both in refinement of previous TIP commands and in the addition of new commands.

We completed design and implementation of the very distant Host interface (see our Quarterly Technical Report No. 12) during the first quarter. The software in the IMP which supports the very distant Host interface is designed for use with the new IMP system (see below) and thus is not yet available in the field. Documentation for the very distant Host interface was completed during the first quarter and will be printed and distributed to the network community early in the second quarter.

Finally, during the first quarter we completed design and began implementation of a new version of the IMP software system. The new version is intended to eliminate the possibility of reassembly lockup and congestion (see our Quarterly Technical Report No. 9 and BBN Report No. 2161, *A Study of the ARPA Network Design and Performance*), to improve the IMP-to-IMP acknowledgment procedure, and to reduce IMP table space requirements. This new system is described in Section 4. At the completion of the system design, we conducted a seminar describing this new system for interested members of the network community late in the first quarter. We expect to install the new IMP software system in the field in the second quarter of 1972.

2. TIP MAGNETIC TAPE OPTION

As one method of increasing the usefulness of the Terminal IMP, we have developed a magnetic tape transfer capability as a TIP option. The first such option was delivered to the field during the first quarter of 1972. In order to ease the problems of interfacing such a specialized terminal type, we chose to specify the attachment of a standard Honeywell peripheral unit rather than attempting to solve the problem of tape drive attachment in a more generalized way. The unit chosen is the Honeywell 316-4021 option which consists of a tape drive controller and one drive unit (the controller itself is capable of handling up to seven additional 316-4022 drives). The characteristics of the tape drive include:

- Read/write speed of 26 inches/second
- Seven track tapes
- Even or odd parity (program selectable)
- Industry compatible 200, 556, or 800 bpi

In addition to the tape drive and controller, the problems of programming for the controller and the buffering of tape records dictated the addition of a separate 4K memory bank to TIPs equipped with this option. The controller also required expansion of the TIP into an additional (lo-boy) cabinet.

The most immediate pressure for the addition of a magnetic tape option to the Terminal IMP was the desire to enable a pair of TIP users to copy tapes over the network from one TIP to another, rather than shipping physical tapes by mail. However, it was clear that, if possible, magnetic tape data should be transmitted according to the Network Working Group's proposed

Data Transfer Protocol (as specified in RFC #264) in order to facilitate tape transfers to other Hosts, and this is the implementation strategy which was adopted.

The magnetic tape system communicates with the network through the TIP, although in many cases it bypasses the usual TIP code, substituting its own procedures to allow for the special nature and relatively high data rate of a magnetic tape terminal. In most respects, however, the tape unit appears as a standard terminal, arbitrarily designated number 63. Thus, on a TIP equipped with magnetic tape, line 63 cannot be used for an external terminal.

An additional terminal is required to issue commands to the tape and receive status information and error comments. This may be of any type and may be connected to any line. Its use as the tape controlling terminal can be concurrent with its normal usage.

The specific hardware design of the magnetic tape units used dictates some constraints. Tape format is 7-track using either odd or even parity. In memory, tape frames are stored in 8-bit bytes, the data bits of each frame occupying the low order six bits of each byte. *Frames can only be written in pairs*; reading a record with an odd number of frames causes the control unit to append an extra null frame to the record in memory.

The maximum record length is 2400 characters (frames). This limit is based on the amount of TIP core available for buffering. If all maximum length records are used, this results in an 80% utilization of tape space at 800 bpi (the remainder is inter-record gaps).

The commands relating to magnetic tapes are in the form of standard TIP commands:

@ MAG SPACE RECORD n
@ MAG SPACE FILE n
@ MAG BACKSPACE RECORD n
@ MAG BACKSPACE FILE n
@ MAG UNLOAD - rewinds tape to load point
@ MAG READ RECORD n
@ MAG READ FILE n
@ MAG WRITE TAPE
@ MAG WRITE EOF - writes a file mark
@ MAG SETUP COPY - establishes "standard" socket numbers

where n is an optional positive integer denoting the number of records or files to be spaced, backspaced, or read. If n is absent, it is defaulted to one. A file mark is treated by the hardware as a record and must thus be accounted for when spacing or reading by the RECORD commands. The SETUP COPY command is used in the establishment of a connection between TIPs, described below. Magnetic tape commands may be stacked; that is, additional commands may be entered for later execution before the current command is completed.

There are some important things to note about magnetic tape commands. All regular TIP commands given for the tape, e.g., those specifying Host or socket parameters, must be preceded by 63 (as the examples below show). This, of course, captures the tape drive for the terminal giving the commands. All special tape commands (those beginning with MAG), <u>implicitly</u> capture device 63 in the same way. Thus once any terminal issues a

command for device 63 or any MAG command, it has captured the magnetic tape; no one else is permitted to control it until the owning terminal has issued the @63 GIVE BACK command.

A network connection must exist before information may be transferred. A typical sequence of TIP commands which might establish a connection between two magnetic tapes follows: at each TIP, the operator would issue an @63 HOST hn where hn is the Host number of the other TIP. Each would then enter a @ MAG SETUP COPY which establishes socket numbers for the "standard" TIP to TIP magnetic tape connection. Then one side would give a @63 PROTOCOL BOTH which would open the connection. Status information about this connection such as OPEN, DEAD, etc. will be prefaced by MTR and MTT rather than the usual R and T to differentiate magnetic tape activity from other activity of the controlling terminal.

The parity of the read tape is sensed automatically. At the beginning of each READ command, a message is sent to the writing TIP informing it of the parity. Thus if a tape has mixed parities, each section with a different parity should be sent with a separate READ command. The parity information at the writing TIP is invalidated by an UNLOAD and must thereafter be reset.

In order to allow a tape to be written, the user must issue a MAG WRITE TAPE command. Writing is then enabled and any data which arrives over the connection will be written on the tape. The write enabled state is terminated either by a closed connection or by rewinding the tape (UNLOAD).

Errors and abnormal status conditions are detected and notice is given to both ends of the connection, where appropriate messages

are typed out on the controlling terminals. Errors which will be of significance to the operator include:

UNREC ERR — Unrecoverable read or write errors after 10 retries — a bad spot in the tape or tape drive hardware problems.

EOT — The tape has moved past the end of tape marker. This does not invalidate the data, but the tape is in imminent danger of slipping off the reel and usually requires a switch to another volume.

The error messages are preceded by MTR or MTT to denote which side of the connection originated the message.

The magnetic tape system transfers information according to the proposed Data Transfer Protocol (DTP), using the Descriptor and Counts (D&C) mode. A D&C data transaction contains tape data, one six bit frame, right-justified, in each eight bit byte. It is assumed that all transactions have an integral number of bytes. Each record will always contain an even number of frames as mentioned above. The concept of DTP transactions transcends those of message and packet; thus there is no enforced relationship between transactions and their start and end with respect to messages. Note that a maximum length record is about 2-1/2 messages long.

Information separators may delimit either records or files. A file separator is sent whenever a file mark is read from the tape. Although the file mark is a record by itself, a separate record separator is <u>not</u> sent to delimit the file separator; the next transaction should properly be the data of the first record in the next file. Each data record always has its own record separator, independent of the file separators.

The D&C control transaction type is presently used only to send parity information. It has one word (16 bits) of data which contains either a zero (even parity) or a 100_8 (odd parity).

The sequence number option is utilized, with D&C data, D&C control, and information separators all deriving their numbers from a common sequence. A received sequence number out of sequence that is not -1 causes an error.

The initial handshaking in a magnetic tape connection proceeds as follows: the existing TIP procedures establish a network connection, perhaps as outlined above. The tape system notices that the connection is open and sends out a small initial allocate, sufficient to allow for a DTP Modes Available transaction. When a sufficient allocation is received from the other end of the connection, a Modes Available is sent out advertising D&C control and data types. If a Modes Available is received which includes these modes, an allocation is returned for five messages and enough bits to allow for a maximum size record. Writing or reading can then commence. If a read parity error is encountered, the parity of the read instruction is changed. The record is reread until successful or until the retry count is reached, which signals an error. When the first record of a READ command has been successfully read, but before it has been sent out, a D&C control message is sent containing the parity of that record. This information is used to determine the write parity on the receiving end. Until the parity information is received, no data will be written on the tape. The write parity is invalidated by an UNLOAD.

As each message is received, a one-message/no-bits allocate is sent. When a tape buffer is freed by writing its contents to tape, the total bits freed plus an amount equal to the number of

bits received in control messages is allocated. Thus the long term, quiescent buffer allocation should remain constant.

When a closed connection is detected, the tape routines are initialized, and all parameters and modes return to their default setting.

When errors are detected, a DTP error transaction is sent to the other TIP, informing it of the error. Both TIPs then output an appropriate message to the controlling terminal, using the existing TIP error facilities, although drawing from a special pool of magnetic tape messages. Errors other than those mentioned above relate to violations of the DTP such as errors in the sequence numbering, bad transaction type (usually a synchronization problem), and illegal parity. These should not occur in a healthy system and are included to aid in debugging failures and new systems.

3. IMP BUFFERING REQUIREMENTS FOR SPECIAL CIRCUITS

During the past several months there has been increasing interest in connecting IMPs to a variety of communication circuits other than the "standard" 50 kilobit/second inter-IMP phone lines. This interest is manifested by such developments as the design of the very distant Host interface, the installation of a short 230.4 Kbs circuit in the network, and serious discussion of expansion of the network to overseas points via satellite circuits or undersea cables with many repeater stations. In the past we have been able to assume that inter-IMP communication circuits:

- operate at "speed-of-light" over distances not greater than about 3000 miles
- operate at 50 Kbs
- need be fully utilized (kept busy with useful traffic) only when the network is heavily loaded, a condition which arises only when most packets are maximum length (i.e., not interactive traffic).

Packets sent on any inter-IMP circuit must be buffered (for possible retransmission) by the transmitting IMP until acknowledged; the set of assumptions listed above permits us to limit the number of buffers provided for any circuit to a maximum of eight and still insure that the line will be fully utilized.

Introduction of "special" circuits, with quite different parameters, into the network is likely to require changes in buffer allocation if the circuits are to be kept busy. For this reason we undertook a study of the relationships among buffer requirements, line speeds, and line lengths during the first quarter. The results of this study are presented below.

The number of buffers required to keep a phone line (or other circuit), busy is a function not only of line bandwidth and distance but also of packet length, IMP delay, and acknowledgment strategy. In order to compute the buffering needed to keep a line busy, we need to know the length of time the sending IMP must wait between sending out a packet and receiving an acknowledgment for it. If we assume no line errors, this time is the sum of:

P_P - Propagation delay for the packet
(time for the first bit to traverse the line)

T_P - Transmission time for the packet
(time to send out all the bits on the line)

L - Latency in the other IMP
(time before an acknowledgment can be sent out)

P_A - Propagation delay for the acknowledgment

T_A - Transmission time for the acknowledgment

The number of buffers we need is then given by:

$$B = \frac{P_P + T_P + L + P_A + T_A}{T_P} \qquad (1)$$

Propagation delays $P_P = P_A = P$ are a simple function of distance. Transmission delays T_P and T_A are proportional to packet length and inversely proportional to line bandwidth. Latency L is a function both of program delay in the other IMP and of the delay caused by having a partially-transmitted packet on the line at the time when the acknowledgment is ready to be sent. The program

delay is essentially zero; thus latency is a function of packet length. Using these relationships, equation (1) can be rewritten as:

$$B = \frac{2P}{T_P} + \left(1 + \frac{L + T_A}{T_P}\right) \qquad (2)$$

That is, the number of buffers needed to keep a line full is proportional to the length of the line and its speed, and inversely proportional to the packet size, with the addition of a constant term. We now introduce two new terms, T_S and T_L, for the transmission times for the shortest and longest packets permitted in the system.

There are three variables we can express in terms of T_S and T_L:

We can make any of the following assumptions about packet length:

$T_P = T_S$ (3a) all short packets

$T_P = T_L$ (3b) all long packets

$T_P = \frac{xT_S + yT_L}{x + y}$ (3c) any mix of short and long packets

We can make either of two assumptions about the latency:

$L = \frac{T_S + T_L}{4}$ (4a) "average" latency

$L = T_L$ (4b) worst case latency

The expression for "average" latency assumes that 1/2 of an "average" packet remains to be sent before the transmission of the acknowledge begins; an "average" packet* is of length $\frac{T_S + T_L}{2}$.

The worst case latency assumes the acknowledge becomes ready for transmission just as the first bit of a maximum length packet is sent.

We can use either of two acknowledgment schemes:

$$T_A = T_S \qquad \text{(5a)} \quad \text{separate acknowledges}$$

$$T_A = \frac{T_S + T_L}{2} \qquad \text{(5b)} \quad \text{"piggyback" acknowledges}$$

Separate acknowledges correspond to the acknowledgment scheme used in the current system. The "piggyback" acknowledgment* scheme is the method which is used by the very distant Host interface and which will be used by the new IMP system (see Section 4).

Several of the terms appearing in these equations are either known parameters for the ARPA network or are functions of physical constants.

Propagation delay is essentially speed-of-light times distance. Some typical network distances and the associated values for P are:

Distance	P	
10 mi.	54 μsec.	typical current line distances
100 mi.	540 μsec.	
1,000 mi.	5.4 msec.	
3,000 mi.	16.2 msec.	cross-country line
10,000 mi.	54 msec.	
45,000 mi.	272 msec.	satellite link

*Note that these expressions assume an output traffic mix at the "receiver" end of the line of half short and half long packets. Variations in this mix have only second order effects on B.

Every IMP packet has:

72 bits of hardware overhead
80 bits of software overhead
0-1008 bits of data.

Therefore, the packet size runs from 152 bits to 1160 bits. Using these values as the lengths of short and long packets, and using standard circuit bandwidths, we can compute typical values for T_S and T_L as follows:

	Circuit Bandwidth			
	9.6 Kbs	50 Kbs	230.4 Kbs	1.4 Mbs
T_S	15.7 msec.	3.04 msec.	660 µsec.	106 µsec.
T_L	120.5 msec.	23.2 msec.	5.03 msec.	812 µsec.

We have used these values for P, T_S, and T_L to compute B from equation (2) using all possible combinations of choices for the following variables:

- Packet length mixes, in terms of equation (3c) of
  - x=1, y=0 (all short)
  - x=8, y=1 (mostly short)
  - x=2, y=1
  - x=1, y=1
  - x=0, y=1 (all long)
- Line lengths of 1, 10, 100, 1000, 10000, and 45000 miles
- "Average" latency and worst case latency
- Separate acknowledges and "piggyback" acknowledges
- Circuit bandwidths of 9.6, 50, 230.4, and 1400 Kbs

The computed values for B are presented in Tables 1 through 4. In addition, curves showing the relationships among the other variables for worst case latency and "piggyback" acknowledgments (corresponding to Table 1) are presented in Figure 1.

9.6KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	12.95	12.95	13.02	13.63	19.74	43.51
8S:1L	7.88	7.88	7.92	8.27	11.79	25.47
2S:1L	4.72	4.72	4.74	4.93	6.84	14.24
1S:1L	3.77	3.77	3.78	3.93	5.34	10.85
0S:1L	2.57	2.57	2.57	2.65	3.46	6.57

50KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	12.95	12.98	13.30	16.48	48.32	172.12
8S:1L	7.88	7.90	8.08	9.92	28.24	99.52
2S:1L	4.72	4.73	4.83	5.82	15.74	54.30
1S:1L	3.77	3.78	3.85	4.59	11.96	40.65
0S:1L	2.57	2.57	2.61	3.03	7.20	23.42

230.4KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	12.96	13.11	14.58	29.25	175.94	746.39
8S:1L	7.89	7.97	8.82	17.26	101.72	430.17
2S:1L	4.73	4.77	5.23	9.80	55.49	233.17
1S:1L	3.77	3.81	4.15	7.54	41.53	173.71
0S:1L	2.57	2.59	2.78	4.70	23.92	98.67

1400KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	13.05	13.94	22.85	111.99	1003.33	4469.65
8S:1L	7.94	8.45	13.58	64.90	578.10	2573.86
2S:1L	4.75	5.03	7.81	35.57	313.20	1392.88
1S:1L	3.79	4.00	6.06	26.72	233.25	1036.42
0S:1L	2.58	2.70	3.86	15.54	132.34	586.55

TABLE 1

Buffer Requirements Assuming Worst Case Latency and "Piggyback" Acknowledges

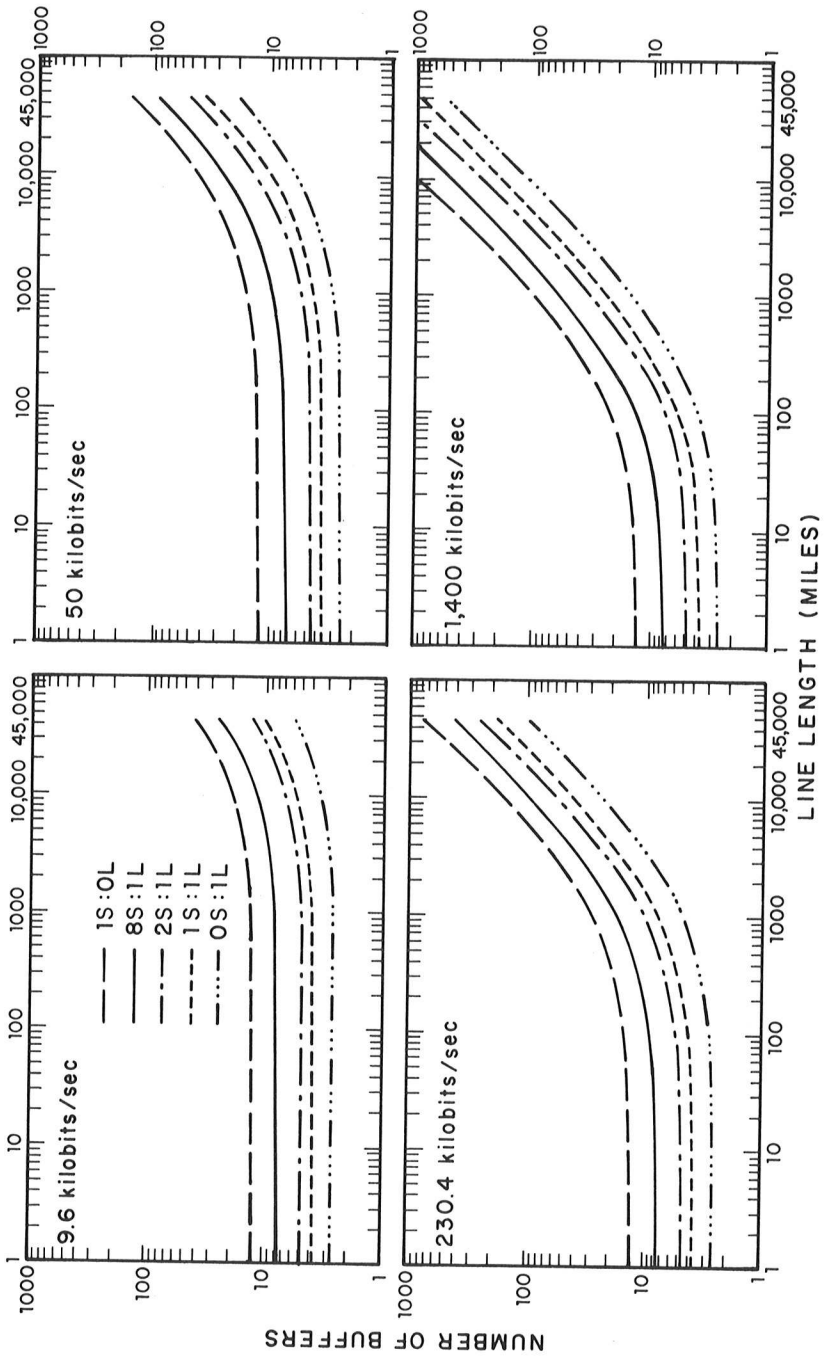

FIG. 1 NUMBER OF BUFFERS vs LINE LENGTH FOR 4 LINE SPEEDS
All Graphs Assume Worst Case Latency and "Piggyback" Acknowledges
Traffic Mixes Shown as Ratio of Short Packets (S) to Long Packets (L)

9.6KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	3.19	3.19	3.26	3.87	9.98	33.75
8S:1L	2.26	2.26	2.30	2.65	6.17	19.85
2S:1L	1.68	1.68	1.70	1.89	3.80	11.20
1S:1L	1.51	1.51	1.52	1.66	3.08	8.59
0S:1L	1.29	1.29	1.30	1.38	2.18	5.29

50KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	3.32	3.35	3.67	6.85	38.68	162.48
8S:1L	2.33	2.35	2.53	4.37	22.70	93.97
2S:1L	1.72	1.73	1.83	2.82	12.74	51.30
1S:1L	1.54	1.54	1.62	2.36	9.73	38.42
0S:1L	1.30	1.31	1.35	1.77	5.94	22.16

230.4KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	3.88	4.03	5.50	20.17	166.86	737.31
8S:1L	2.66	2.75	3.59	12.04	96.49	424.94
2S:1L	1.90	1.94	2.40	6.97	52.66	230.34
1S:1L	1.67	1.70	2.04	5.44	39.43	171.61
0S:1L	1.38	1.40	1.59	3.51	22.73	97.48

1400KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	7.57	8.46	17.38	106.51	997.85	4464.18
8S:1L	4.78	5.30	10.43	61.75	574.95	2570.71
2S:1L	3.05	3.32	6.10	33.86	311.50	1391.17
1S:1L	2.52	2.73	4.79	25.45	231.98	1035.15
0S:1L	1.86	1.98	3.15	14.83	131.62	585.83

TABLE 2

Buffer Requirements Assuming "Average" Latency and "Piggyback" Acknowledges

9.6KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	8.64	8.65	8.71	9.32	15.43	39.20
8S:1L	5.40	5.40	5.44	5.79	9.31	22.99
2S:1L	3.38	3.38	3.40	3.59	5.49	12.90
1S:1L	2.77	2.77	2.79	2.93	4.34	9.85
0S:1L	2.00	2.00	2.01	2.09	2.89	6.01

50KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	8.67	8.70	9.02	12.20	44.04	167.84
8S:1L	5.42	5.43	5.62	7.45	25.78	97.06
2S:1L	3.39	3.40	3.50	4.49	14.41	52.97
1S:1L	2.78	2.78	2.86	3.60	10.97	39.66
0S:1L	2.01	2.01	2.05	2.47	6.64	22.86

230.4KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	8.81	8.96	10.43	25.09	171.78	742.24
8S:1L	5.50	5.58	6.43	14.87	99.33	427.78
2S:1L	3.43	3.48	3.94	8.50	54.20	231.88
1S:1L	2.81	2.84	3.18	6.58	40.57	172.75
0S:1L	2.02	2.04	2.24	4.16	23.38	98.13

1400KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	9.73	10.62	19.54	108.67	1000.01	4466.34
8S:1L	6.03	6.54	11.67	62.99	576.19	2571.95
2S:1L	3.72	4.00	6.77	34.54	312.17	1391.84
1S:1L	3.02	3.23	5.29	25.95	232.48	1035.65
0S:1L	2.14	2.26	3.43	15.11	131.90	586.11

TABLE 3

Buffer Requirements Assuming Worst Case Latency and Separate Acknowledges

9.6KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	4.16	4.16	4.23	4.84	10.95	34.72
8S:1L	2.82	2.82	2.86	3.21	6.73	20.41
2S:1L	1.98	1.99	2.00	2.20	4.10	11.50
1S:1L	1.73	1.73	1.75	1.89	3.31	8.81
0S:1L	1.41	1.41	1.42	1.50	2.30	5.42

50KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	4.16	4.19	4.51	7.69	39.53	163.33
8S:1L	2.82	2.84	3.02	4.85	23.18	94.46
2S:1L	1.98	1.99	2.09	3.09	13.00	51.56
1S:1L	1.73	1.74	1.81	2.55	9.93	38.61
0S:1L	1.41	1.42	1.46	1.88	6.05	22.27

230.4KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	4.17	4.32	5.79	20.46	167.15	737.60
8S:1L	2.83	2.91	3.76	12.20	96.66	425.11
2S:1L	1.99	2.03	2.49	7.06	52.75	230.43
1S:1L	1.74	1.77	2.11	5.51	39.50	171.68
0S:1L	1.42	1.44	1.63	3.55	22.77	97.52

1400KB

	1MI	10MI	100MI	1000MI	10000MI	45000MI
1S:0L	4.26	5.15	14.06	103.20	994.54	4460.86
8S:1L	2.88	3.39	8.52	59.84	573.04	2568.80
2S:1L	2.01	2.29	5.07	32.83	310.46	1390.14
1S:1L	1.75	1.96	4.03	24.68	231.21	1034.38
0S:1L	1.43	1.54	2.71	14.39	131.19	585.40

TABLE 4

Buffer Requirements Assuming "Average" Latency and Separate Acknowledges

4. TRANSMISSION AND FLOW CONTROL

We have known for some time that the current version of the IMP system is susceptible to a condition which is called reassembly lockup. Once reassembly lockup has occurred at some IMP, no traffic can flow to that IMP. Although system timeouts can temporarily unlock the network, lockup is virtually guaranteed to recur if the level of traffic remains the same. Even without lockup, congestion can occur under conditions of heavy traffic flow to a single site. These conditions have arisen infrequently only because current network usage is light and because the vast majority of current traffic consists of single-packet messages. Nevertheless, reassembly lockup has occurred and will continue to do so with increasing frequency unless the software system is changed.

During the first quarter we completed the design of a new IMP software system which will prevent reassembly lockup and make congestion extremely unlikely. Implementation of this new system was well under way by the end of the quarter and should be installed in the field during the second quarter. In addition, we have taken the opportunity provided by this major change to redesign the inter-IMP acknowledgement scheme and to make other changes which reduce IMP table space requirements. It is important to note that none of these IMP system changes will require changes to the Hosts' Network Control Programs.

4.1 Flow Control and Lockup Prevention

The link mechanism is an inadequate technique for Host-to-Host flow control. Not only can Hosts "spray" on many links and congest the Network, but they can also cause reassembly lockup, a condition under which no traffic can flow to the destination IMP. This occurs

when reassembly storage at a destination is completely used up by partially reassembled messages and neighboring IMPs fill with store-and-forward packets for that destination. Once this kind of congestion has developed, a lockup occurs when the missing packets for the messages being reassembled are held *two or more* hops away from the destination.

We have developed a method of controlling such congestion which is based on allocation messages sent from the destination IMP to the source IMP. When an IMP has a *multi-packet* message to send, it first sends off a "request for allocation" (of reassembly space) to the destination IMP. Some time later it will receive an "allocate" message and at that point it may proceed to transmit the message. This procedure ensures that the destination is never swamped and that reassembly lockup will not occur. The request/allocate sequence does introduce a certain amount of overhead, however, and we wish to provide as much bandwidth as possible for multi-packet messages. Therefore, we will insure that there is no necessity for the "request for allocation" in the case of a steady stream of traffic. When the destination IMP has given a multi-packet message to its Host, it returns a RFNM to the source and at the same time allocates reassembly storage for the anticipated next message. The source IMP receives a new "allocate" along with the RFNM, and if the source Host is responsive enough (sends again with 125 msec of the time the RFNM is received) the message can be transmitted right away. If the source Host waits too long, or has nothing more to send, the source IMP will return the allocation by means of a "give back" message. The next time the Host tries to send, the IMP will transmit a "request for allocation", and wait for an "allocate" before proceeding.

For *single packet* messages, we are interested in minimizing the delay encountered through the network. The request mechanism used for multi-packet messages would slow down one-packet messages too much. Instead, we will send the one-packet message along with the "request for allocation" and save a copy of the message at the source IMP. If the destination IMP can take the message, it does so immediately, and returns a RFNM to the source. If there is not enough storage at the destination, it sends back an "allocate" message later, when the storage becomes available. When the source receives this allocate, it retransmits the message (without the request indication this time). In this approach, RFNMs are passed along to the source Host as before, but requests and allocates are internal to the IMP sub-network.

4.2 IMP-to-IMP Transmission Control

The goal of IMP-to-IMP transmission control is to detect errors and provide for retransmission if they occur. To this end, cyclic redundancy check hardware has been incorporated into the IMP-modem interfaces for error detection, and a positive acknowledgment/timeout scheme is used for retransmission. The software also provides for detection of duplicate transmissions and/or duplicate acknowledgments if they occur.

In the current system, each acknowledgment is sent as a separate message, and the timeout period is 125 milliseconds (about three times as long as a cross-country round trip). There are two major disadvantages to this scheme:

1) Software (message identification) and hardware (framing and checksum) overhead combine to make each acknowledgment 152 bits long. Thus, although only a

few bits of useful information are being conveyed, acknowledgments consume a significant portion of line bandwidth at times of heavy load.

2) The timeout period for retransmission was made relatively long in order to avoid unnecessary retransmissions, and consequent loss of overall bandwidth, at times of heavy load. On the other hand, at times of light traffic a packet must wait much longer than "necessary" for retransmission, thus reducing both throughput and responsiveness.

For these reasons we have redesigned the acknowledgment scheme to be similar to the very distant Host connection as described in our Quarterly Technical Report No. 12. In this scheme, each physical line is broken into a number of logical "channels", currently eight channels in each direction. Acknowledgments are carried "piggyback" by normal network traffic in a set of acknowledgment bits contained in every packet, thus reducing the bandwidth they require. In addition, the period between retransmissions will become dependent upon the volume of new traffic.

Appended to each packet are several bits of control information including an "odd/even" bit which is used to detect duplicate packet transmissions, a three-bit channel number, and eight acknowledge bits — one for each channel in the reverse direction.

Each of the packets going in one direction is associated with one of the channels mentioned above. For each transmit channel a used/unused bit and an odd/even bit are kept (both initialized to zero). The used/unused bit indicates whether there is currently a packet associated with the channel. For each receive channel,

an odd/even bit is kept (initialized to one). The transmit side cycles through its used channels (those with packets associated with them), transmitting the packets along with the channel number and the associated odd/even bit. At the receive side, if the odd/even bit of the received packet does *not* match the odd/even bit associated with the appropriate receive channel, the receive odd/even bit is complemented, otherwise the packet is a duplicate and is discarded.

Acknowledgments of all packets correctly received at the receive side, whether the acknowledgments are duplicate or not, are sent to the transmit side at the other IMP. This is done by copying the receive odd/even bits into the positions reserved for the eight acknowledge bits in the control portion of *every* packet transmitted. In the absence of other traffic, the acknowledges are returned in "null packets" in which *only* the acknowledge bits contain relevant information (i.e., the channel number and odd/even bit are meaningless; null packets are not acknowledged). When the transmit side receives a packet, it compares (bit by bit) the acknowledge bits against the transmit odd/even bits. For each match found, the corresponding channel is marked unused, the corresponding packet is discarded, and the odd/even bit is complemented.

In view of the large number of channels, and the delay that is encountered on long lines, some packets may have to wait an inordinately long time for transmission. We do not want a one-character packet to wait for several thousand-bit packets to be transmitted, multiplying by 10 or more the effective delay seen by the source. We have therefore instituted the following transmission ordering scheme: first, we send any new priority packets

(see Section 4.3); then any new regular packets; then, if there are no new packets to send, we retransmit previous unacknowledged packets. In addition, the system ensures that unacknowledged packets are periodically retransmitted even when there is a continuous stream of new traffic.

4.3 Host-to-Host Transmission Control

The problem of Host-to-Host communication is somewhat different from the IMP-to-IMP situation outlined above. There may, of course, be many IMPs in the transmission path between Hosts. We introduced the technique of breaking Host messages into packets to minimize the delay seen for long transmissions over many hops. These packets may arrive at the destination out of order, and in the event of a broken line or IMP, there may be duplicate packets. The reassembly logic in the destination IMP currently performs the task of reordering the packets and culling duplicates, waiting until all the packets have arrived and only then passing them on to the destination Host and returning a RFNM to the source. Sequential message numbers are assigned to each transmission on each link in order to detect and discard packets from messages other than the current one. This strategy is based on the rule that on each link between Hosts only one message may be in transmission.

We wished to change this strategy in the following ways:

1. It should be possible to have more than one message in transit between a pair of processes. Currently, a Host could use more than one link to achieve this effect, "spraying" the transmissions from one process on many links. It seems that this is not the right way to use links; they should be used for (and Host/

Host protocol uses them for) multiplexing connections to the various processes in a Host. The IMP does not control the number of links in use, except to set an upper bound, but this lack of control can lead to congestion problems. Therefore, we decided that the old function of the message number would be expanded to include the function of ordering Host-to-Host transmissions.

Specifically, we will allow up to four messages to be in transmission from a source IMP to a destination IMP. All the source and destination Hosts share this message space. There is a message number assigned to each transmission at the source, and the destination has a "window" of four acceptable message numbers out of a message-number space of 256. Messages with out-of-range message numbers are discarded, as well as duplicate messages and duplicate packets. RFNMs are returned for message numbers, and the IMP will no longer perform any bookkeeping associated with link numbers. The message number is an internal device to order messages into the destination Host, and the link number is a separate external code which the IMP merely passes along as data.

2. We also wished to allow for a priority path between Hosts, to bypass the regular message ordering scheme. That is, there should be a second path between Hosts in which messages can flow independent of the regular path, and when the next message on either path is ready that message is delivered to the Host. We will implement this dual-path scheme by making a bit in the Host-to-IMP leader available to the Hosts as a "priority flag". The source IMP will associate a priority-sequence number with each priority message; the assignment of priority-sequence numbers is cyclic through a range of four numbers, with each source-IMP/destination-IMP pair cycling independently (as with

message numbers). At the destination, a priority message whose priority-sequence number is "next for delivery" will be delivered to the Host even if its message number is not next in line for delivery.

For example, suppose that some (source) IMP A is ready to assign message number 13 and priority-sequence number P2 for (destination) IMP B. Suppose that it receives a sequence of four messages for destination B, with the second and third messages flagged as priority messages. IMP A will assign these messages the numbers 13, 14-P2, 15-P3, and 16. The order of delivery to the Host at B will depend on the order of arrival at IMP B as shown below:

Arrival Order at IMP B	Delivery Order by IMP B
13, 14-P2, 15-P3, 16	13, 14-P2, 15-P3, 16
15-P3, 16, 14-P2, 13	14-P2, 15-P3, 13, 16
14-P2, 13, 16, 15-P3	14-P2, 13, 15-P3, 16

In other words, a message cannot be delivered to its destination until either its message number or its priority-sequence number is "next for delivery", but the priority mechanism allows some messages to "leapfrog" ahead of their position within the message number assignment.

3. In addition to the window of acceptable message numbers that the source and destination IMPs maintain, there is a set of bits corresponding to outstanding messages. The source IMP keeps track of whether a response has come in for each message (such as a RFNM or other control message), in order to detect duplicate responses. The destination IMP keeps track of whether the message is complete (whether all the packets have arrived)

in order to detect duplicate transmissions. The source IMP also times out the message number, and if a response has not been received for some message within 30 seconds, the source IMP sends out a control message with the timed-out message number, questioning the possibility of an incomplete transmission. The destination IMP must always return a RFNM for such a message, stating whether it saw the original message or not, and the source IMP will inquire every 30 seconds until it receives a response. This technique allows the source and destination IMPs to be synchronized in the event of a lost message or RFNM. It should be noted that this kind of failure is very infrequent, and happens only when an intermediate IMP fails to run its program correctly for some reason.

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Mass. 02138

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b. GROUP

3. REPORT TITLE
QUARTERLY TECHNICAL REPORT NO. 13

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*
Bolt Beranek and Newman Inc.

6. REPORT DATE
April 1972

7a. TOTAL NO. OF PAGES
31

7b. NO. OF REFS

8a. CONTRACT OR GRANT NO.

b. PROJECT NO.

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)
BBN Report No. 2353

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. DISTRIBUTION STATEMENT

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
Advanced Research Projects Agency
Arlington, Virginia 22209

13. ABSTRACT

The basic function of the ARPA computer network is to allow large existing computers (Hosts), with different system configurations, to communicate with each other. Each Host is connected to an Interface Message Processor (IMP), which transmits messages from its Host(s) to other Hosts and accepts messages for its Host(s) from other Hosts. There is frequently no direct communication circuit between two Hosts that wish to communicate; in these cases intermediate IMPs act as message switchers. The message switching is performed as a store and forward operation. The IMPs regularly exchange information which: allows each IMP to adapt its message routing to the conditions of its local section of the network; reports network performance and malfunctions to a Network Control Center; permits message tracing so that network operation can be studied comprehensively; allows network reconfiguration without reprogramming each IMP. The Terminal IMP (TIP), which consists of an IMP and a Multi-Line Controller (MLC), extends the network concepts by permitting the direct attachment (without an intervening Host) of up to 64 dissimilar terminal devices to the network. The Terminal IMP program provides many aspects of the Host protocols in order to allow effective communication between a terminal user and a Host process.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Computers and Communication						
Store and Forward Communication						
ARPA Computer Network						
Interface Message Processor						
IMP						
Terminal IMP						
TIP						
Honeywell DDP-516						
Honeywell DDP-316						
Multi-Line Controller						
MLC						
Network Control Center						
Host Protocol						
High Speed Modular IMP						
HSMIMP						
Reliable Transmission Package						