Connecting terminals to multiple LANs*

by BRONSON HOKUF, PAUL D. AMER, and DANIEL GRIM

University of Delaware
Newark, Delaware

ABSTRACT

With computing facilities often distributed over sites that are both physically and administratively separate, many organizations find themselves with several unconnected Local Area Networks (LANs). Eventually it becomes desirable to enhance communication capabilities between these LANs so that users can access all hosts on all LANs. This paper describes a terminal gateway, a system that provides users having terminal access to hosts on one LAN with an ability to access all hosts on another LAN. Such a gateway solution is appropriate when: (1) separate terminal connections to each LAN are prohibitively expensive and (2) commercially available bridges or gateways are unsuitable for use. A functional specification of the terminal gateway, the communication protocol it uses, and a specification of performance metrics maintained by the gateway are described.

* This research was supported in part by the Office of Naval Research Department of Defense Contract Number N00014-83-K-0320
INTRODUCTION

Local area networks (LANs) are clearly recognized as an economical means of connecting computing resources. With computing facilities often distributed over sites that are both physically and administratively separate, many organizations find themselves with several unconnected LANs. Eventually it becomes desirable to enhance communication capabilities between these LANs so that users can access all hosts on all LANs. Typical solutions to this problem include: combining the LANs into one LAN by using repeaters, attaching a bridge or gateway between pairs of networks, or providing separate terminal connections to hosts on the different networks. A terminal's physical connections may be extended by installing additional private lines, leasing lines, multiplexing many terminal lines over leased, or private lines, or by providing some sort of internetwork connection to allow host access using existing terminal lines.

Two LANs exist at the University of Delaware, one connecting computing center machines, ACSnet, for instructional purposes, and one connecting machines in a joint EE and CIS Department laboratory, EE/CISnet, for research purposes. Terminal connections to the ACSnet are provided via leased serial lines to a port selector. (A port selector is an intelligent switching device which establishes a virtual connection between a user's terminal line and the host computer for which they have requested a connection.) (See Figure 1.) Terminal connections to the EE/CISnet are provided by university owned lines to a terminal switch on the EE/CIS Ethernet. Users that needed a connection to both networks had two lines to their terminal and a locally developed switchbox which allowed them to switch between the two lines. With the recent tripling of costs for leased lines, an alternative connection to the ACS hosts supported by the port selector process was needed to eliminate the leased lines while still providing satisfactory access to hosts on both networks. Several solutions were considered; the one selected and discussed in this paper was the development of a terminal gateway.

A terminal gateway is not a general solution to inter-networking multiple LANs such as one may obtain with bridges for homogeneous LANs or general gateways for heterogeneous LANs. A terminal gateway provides more economical terminal connections to other LANs for users already connected to one LAN. More general interLAN problems such as file transfer and distributed computing (e.g., load sharing) require higher level protocols for host-to-host communication and are outside the scope of this solution.

The solution presented, as it was implemented at the University of Delaware, should be applicable to other sites that seek to provide terminal connections to hosts on a distant LAN when they already have connections to an existing LAN and physically joining the LANs into one is not feasible. The computing environment serviced is discussed in the Background and Overview of the Problem section, along with the motivation for considering a terminal gateway solution. The section on the terminal gateway presents the hardware configuration, and reviews the operation and performance of the terminal gateway. This includes the gateway's functional specification, communication protocol and self-monitoring capability. Finally, the last section, concludes that a terminal gateway is a technically feasible and economically attractive solution to providing terminal access to multiple LANs.
BACKGROUND AND OVERVIEW
OF THE PROBLEM

University of Delaware Computing Environment

Local networking efforts at the University of Delaware are typical of a campus with growing communication capabilities. Together, the Electrical Engineering (EE) and Computer and Information Sciences (CIS) Departments maintain a joint computing laboratory for research purposes consisting of several Vaxes and other machines linked together by an Ethernet (EE/CISnet). (See Figure 1.) In addition, the University's Academic Computing Service maintains a similar Ethernet (ACSnet) interconnecting Vaxes, a Pyramid 98XE, and an IBM 3081D for general instructional computing and non-EE/CIS faculty research.**

The main hardware on the EE/CISnet are seven Vax-11s, four running 4.2 BSD UNIX, and three running VMS. Two Vaxes are connected to the ARPAnet via IMP*** 96 providing a major ARPAnet connection point in the northeast corridor. Two Bridge Communication GS-3 half-gateways directly connect the EE/CISnet to the ACSNet over a 64 Kbit/s leased line. This provides a functional LAN-LAN interconnection for file transfer (e.g., ftp) and remote login (e.g., telnet or rlogin). User connections to the Vax systems are accomplished by a PDP-11/45 acting as an intelligent terminal switch. Derived from software originally developed at Cornell University, the switch physically connects up to 112 terminals and provides multiple, simultaneous virtual terminal connections to one or more hosts for each terminal. The gateway that is the subject of this paper resides on the EE/CISnet and allows terminals attached to the EE/CISnet terminal switch to access the port selector of the ACSnet.

Motivation for a Terminal Gateway

The desire to implement a gateway for terminal traffic came as a direct result of the divestiture of AT&T and the resulting increases in communication costs. The CIS Department maintains approximately 31 leased public telephone network lines for terminal connections to the ACSnet. The cost of these connections has multiplied dramatically over a short period of time. The Department had to find a way to reduce, if not eliminate, the prohibitive leased line costs while still providing access to all hosts on both LANs. In addition, having users deal with two physical lines to their terminal was undesirable for at least the following reasons. First, switching between the two networks, by means of a locally developed switchbox, usually meant reconfiguring the terminal for a different baud rate. Second, they could not maintain the first connection and simultaneously set up a connection to the other network. Third, the extra hardware involved meant extra maintenance costs.

Alternative solutions

Several alternatives, other than the terminal gateway (TGW) that was chosen, were discussed and considered. One option was to combine the existing Ethernet into a single Ethernet connecting the two sites. This option was complicated by the legalities of not owning a right-of-way for such a connection. In general, even if a right-of-way exists, one may not be able to extend the LAN due to exceeding the maximum end-to-end length restriction. In Delaware's case, products existed, such as optical repeaters, DEC's LAN Bridge 100, that appeared able to provide the desired service.*** As an added but uncommon complication, the two LAN's are managed by separate organizations and any solution could not require major LAN changes. Repeaters were eliminated because we could not expect the ACSNet to support the software from the Terminal Switch. In addition, because EE/CISnet is an experimental network, it was undesirable to risk corrupting the ACSNet with experimental traffic; this was a real problem since repeaters forward all traffic between nets.

A related solution was to use the existing Bridge Communication half-gateways**** not only for periodic host-to-host ftp or telnet connections, but for all terminal connections to the ACSNet. The gateways are intelligent enough to isolate the traffic on the two networks, but one still would need either to: (1) port the terminal switch (TS) software to the ACSNet hosts or (2) run a telnet (or rlogin) connection for every active user. The problem of porting the TS host software is complicated by the independent management of the two networks and the fact that one of the ACS hosts, the IBM 3081D, does not support UNIX; the TS host software makes extensive use of UNIX facilities.

The problem, of using telnet connections over the gateways for all active ACSNet users, is this user level process consumes additional host resources at both source and destination, wasting a significant number of CPU cycles on the hosts which could be used for research purposes. Telnet connections require more resources from the remote host than equivalent serial line connections do. Additionally, telnet requires a local host to support both a TS connection and a telnet connection since the terminal must attach to the local host using the TS and the local host communicates with the remote host using the telnet protocol, neither of these are required of a host for a simple serial connection as in the TGW solution. Also, telnet style connections do not always provide the same level of service that a serial line might be expected to provide, for example, normal telnet connections do not support full screen terminal service on the IBM machines. The IBM machine

*** Repeaters are devices used to interconnect cable segments within a single local area network. These devices operate at the physical layer and perform no filtering on the frames received; every received frame is forwarded. A bridge has added functionality and interconnects two distinct LANs. Bridges operate at the datalink layer and are relatively simple, but intelligent, filtering devices whose function is to store-and-forward frames between two local area networks that use the same protocols.3,9,11

**** A gateway is a more complex device, which operates at the network layer and is designed for interconnecting heterogeneous networks. It has the ability to accommodate differences between the two networks being connected, for example: different addressing schemes, packet sizes, network speeds, as well as other differences.4
normally interfaces with terminals through a 7171 Protocol Converter (see Figure 1), which enables many terminals to emulate an IBM 3270 terminal. When the connection to the IBM occurs through the network interface this protocol conversion function is not available. The emulation provided by the protocol converter must be provided by some other source, imposing an additional burden on one of the participating hosts. Additionally, using telnet would compete with mail and ftp for the resources of the GS-3 half-gateways, potentially degrading their performance. While telnet and rlogin are certainly usable on a small scale, they were eliminated as an option due to the expected large overhead they would incur and the unsatisfactory performance for connections to the IBM host.

Another option was to place a local MUX in the areas where terminals are concentrated, run private lines from the terminals to the MUX, and connect the MUX to the remote site via fewer high-bandwidth leased lines. This solution would reduce communication costs, however it required extensive rewiring, complicated by the wide and varying distribution of the terminals requiring service.

Other options included setting up a satellite link, a microwave link, or installing a wide area network interconnecting all university sites. However, these solutions appeared too expensive and their realization too far in the future to be considered as viable candidates for solving the present problem.

The option decided upon was to build a terminal gateway from the EE/CISnet to the ACSnet. A TGW makes use of the existing private connections to attach terminals to the EE/CIS terminal switch and to use a 32 channel statistical MUX operating over a 64 Kbit/s leased line for the link to the ACSnet port selector. By making use of existing, privately owned terminal lines to the EE/CISnet and providing a mechanism for each terminal to connect to the ACS Port Selector via these lines, the leased line costs could be eliminated and a serial connection to an ACS host could be provided. Some experience had been gained in this type of application through the construction of the TS. The TS handles multiple, serial connections directed onto the Ethernet and the TGW would handle multiple, serial connections going off the Ethernet (see Figure 1).

The TGW solution seemed to hold the most promise for a fairly rapid solution with a minimal outlay of additional funds and labor. The hardware was available within existing department resources and no rewiring was necessary. It also provided the additional benefit of giving users a uniform connection interface to all computer systems.

### Cost/benefit analysis

The cost of communications for the CIS Department goes much beyond the costs listed in this section; only the portions relevant to this discussion are listed. One can see from Table I that there has nearly been a tripling in the monthly cost of leased phone lines in the past 3 years. In contrast to the cost of leased phone lines, the TGW presents a more stable and affordable approach to providing the same services. Because the distribution of the available lines would be dynamic,

<table>
<thead>
<tr>
<th>Year</th>
<th>Monthly Line Cost</th>
<th># of Lines†</th>
<th>Total Cost/yr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>$24</td>
<td>27</td>
<td>$7,776</td>
</tr>
<tr>
<td>1984</td>
<td>$28</td>
<td>29</td>
<td>$9,744</td>
</tr>
<tr>
<td>1985</td>
<td>$66</td>
<td>31</td>
<td>$24,552</td>
</tr>
<tr>
<td>1986*</td>
<td>$69</td>
<td>33</td>
<td>$27,324</td>
</tr>
<tr>
<td>1987*</td>
<td>$73</td>
<td>35</td>
<td>$30,660</td>
</tr>
<tr>
<td>1988*</td>
<td>$76</td>
<td>37</td>
<td>$33,744</td>
</tr>
<tr>
<td>1989*</td>
<td>$80</td>
<td>39</td>
<td>$37,440</td>
</tr>
<tr>
<td>1990*</td>
<td>$84</td>
<td>41</td>
<td>$41,328</td>
</tr>
</tbody>
</table>

* Costs projected @5% annual increases.
† Projected to increase two lines per year.

A smaller total number of connections are able to service a larger user group. The TGW solution could be implemented without a major outlay of funds and could be expected to produce substantial savings over the course of the next five years (see Table II). The cost of the TGW was arrived at by computing the replacement cost of the hardware involved in the building of the TGW. (See the hardware description later in the paper.) It also included the cost of the MUXes and modems which were purchased for the project. It did not include the cost of software development, which was funded through academic research, and did not include any amount for maintenance or replacement of the hardware. Maintenance is normally handled by the lab staff and spare hardware is available.

One can see from Table II that the difference in cost between the two approaches is significant. The rise in communication costs was significant enough to motivate a variety of cost containment procedures in the university’s communication services. The TGW is but one of the attempts to contain such costs.

### Problems Associated with a TGW

The TGW solution is not without problems; one need only look at the data path to see the potential for communication bottlenecks (see Figure 2). The TGW must provide a reliable, real-time interface for the user. While the level of service provided cannot be expected to be the same as when each user

<table>
<thead>
<tr>
<th>Method</th>
<th>5 Year (1986–1990) Total*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple leased lines</td>
<td>$170,496</td>
</tr>
<tr>
<td>Terminal gateway†</td>
<td>$22,278</td>
</tr>
<tr>
<td>Projected savings</td>
<td>$148,218</td>
</tr>
</tbody>
</table>

† Includes cost of leased lines and replacement costs of hardware.

Replacement costs from: Midwest Systems, Inc. Burnsville, MN 55337

* (@5% annual increases)
has a dedicated 9600 baud line, the interactive nature of the traffic requires that a rapid response to user requests be given, for example, character echoes must appear to be nearly instantaneous. The TGW adds another TS-like device (which introduces delay) into the data path, plus it multiplexes 32 connections over a 64 Kbit/s link instead of a 10 Mbit/s link. Because of this, the TGW performance is expected to be poorer than the service provided by a dedicated line. However, because communication through the TGW is limited to user generated, interactive traffic, the performance is expected to be satisfactory.

Another problem is the limited number of channels available; if a thirty-third person attempts to use the link, that person is unable to do so. In contrast, when users own physical connections to the port selector, they are guaranteed a connection whenever the port selector has one to give to them.

THE TERMINAL GATEWAY

How the TGW fits into the International Standards Organization's (ISO) Reference Model of Open Systems Interconnection (OSI) is now discussed. Then the design requirements of the TGW are presented followed by a description of the operation of the TGW and its associated hardware.

One possible approach to providing gateway services is to use a standard protocol for all internetwork traffic regardless of the type of networks being connected. This is the approach taken by the ARPAnet Internet Protocol (IP) which defines the format of internet packets and the rules for performing internet protocol functions based on the information in the internet packet headers. IP provides a datagram service and fits between the network (routing) and transport (end-to-end delivery) layers of the ISO-OEI reference model. In addition to IP, ARPAnet gateways use a gateway-to-gateway protocol to exchange routing information. 10

A second, special purpose, approach is to build a gateway that is essentially a protocol converter. The technique is to receive a packet, strip the control information from the packet, and retransmit the data using the protocol of the destination network. The principal disadvantage of this approach is that a different gateway must be built for each different pair of networks interconnected. 11 The loss of generality in this approach tends to make the implementation easier because all of the assumptions are known beforehand. This type of gateway may potentially operate faster (or as fast as, but with less sophisticated hardware) than the more general type of gateway due to the less sophisticated nature of its duties.

The terminal gateway, discussed herein, aligns most closely with the second approach. However, it is important to recall that the TGW is not a LAN-LAN gateway. The TGW does not establish a connection to a remote network, rather it establishes a connection to a port selector which, in turn, establishes a connection to a host on the remote network. It may be helpful to consider the TGW as two distinct parts (although in reality there is only one). The first part resides on the EE/CISnet and, from the point of view of the terminal switch, looks like a gateway to the ACSnet. The TS addresses packets destined for terminal connections on ACSnet to the TGW and receives packets back from the TGW just as if the TGW were the ACS host, the TS treats the TGW the same way it treats a host on its own LAN. The second part of the TGW is attached to a 32 channel, asynchronous, statistical multiplexer. This MUX is attached to an identical MUX at ACS (which is attached to the ACS port selector) over a 64 Kbit/s leased line (see Figure 2). From this side, the TGW appears to behave much like a Packet Assembler/ Disassembler, or PAD, taking the TS packets, disassembling them and sending each piece of data to the appropriate MUX line. The data received from the individual MUX lines is assembled into a packet and sent to the TS over the EE/CIS network. The TGW provides flow control and buffering between the two different transmission media. It does not provide for any acknowledgement or retransmission of packets received.

This last characteristic leaves a possibility of failure. There is no guarantee that a data path is reliable unless there is some sort of end-to-end internetworking protocol with extended functions, 12 usually implemented in the transport layer (e.g., the ARPAnet Transport Control Protocol-TCP). 13 Because of the low bit error rate of the Ethernet, 14 it was chosen to implement the TS-TGW communication protocol as a connection-oriented, sequenced, unreliable protocol. The reliability of the MUX link from the TGW to the ACS port selector is a different matter and is handled by a reliable MUX-to-MUX protocol.

In summary, the TGW functions in a manner analogous to a special purpose gateway which only serves the EE/CIS terminal switch. It makes use of the Ethernet as a backbone, providing virtual circuit connections 15 to users attaching terminals to host computers resident on the remote network.

---

10 Virtual circuit connection implies that the connection behaves as if it were a single dedicated copper circuit, when it is actually implemented in software.
Connecting Terminals to Multiple LANs

TABLE III—Messages processed by the console

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>?,h</td>
<td>HELP</td>
<td>Displays a list of legal console options.</td>
</tr>
<tr>
<td>V</td>
<td>VERBOSE</td>
<td>Displays a list of options with explanations.</td>
</tr>
<tr>
<td>S</td>
<td>SHUTDOWN</td>
<td>Sends a shutdown warning to all attached lines.</td>
</tr>
<tr>
<td>A</td>
<td>ALL</td>
<td>Display all stats kept by the TGW.</td>
</tr>
<tr>
<td>L</td>
<td>LINE</td>
<td>Display status for all MUX connections.</td>
</tr>
<tr>
<td>I</td>
<td>IN-USE</td>
<td>Display line number and idle time for lines in use.</td>
</tr>
<tr>
<td>M</td>
<td>MUX-STATS</td>
<td>Display character counts for each line.</td>
</tr>
<tr>
<td>E</td>
<td>EN-STATS</td>
<td>Display stats for the Ethernet packets</td>
</tr>
<tr>
<td>H</td>
<td>DH-STATS</td>
<td>Display stats for the DH-11 boards.</td>
</tr>
<tr>
<td>Z</td>
<td>DZ-STATS</td>
<td>Display stats for the DZ-11 boards.</td>
</tr>
</tbody>
</table>

Requirements—What the TGW Will Do

The following requirements were established for the initial design of the TGW and they describe the functionality that the TGW provides:

1. The TGW will establish a connection with the ACS port selector upon request from a terminal attached to the TS.
2. The TGW will not introduce unreasonable delay into the system response time (i.e., the delay for echoing a character must remain non-noticeable).
3. The TGW will provide an option to logout idle connections if no characters are transmitted during some fixed period of time.
4. The TGW will accept messages from a console and act upon those messages. Options supported are listed in Table III.
5. The TGW will print certain events on the console device. Those events are: reception of a bad Ethernet packet, with the reason for the packet being bad; disconnection of a physical line to the MUX; refusal of a request for a connection due to no available line; and automatic logout of a line along with the line number that was affected.
6. The TGW will adapt to the failure of a TS, correctly terminating any affected connections.
7. The TGW will dynamically detect and flag MUX lines that are not connected or that become disconnected.
8. The TGW will (optionally) keep statistics on itself; these statistics are listed in Table IV.

Restrictions—What the TGW Will Not Do

There were certain functions the TGW was not to provide. The following restrictions were accepted as part of the TGW specification:

1. The TGW will not introduce any error detection or correction as part of the communication with the TS. No reliability will be added to the Ethernet.
2. The TGW will not be designed to handle applications (e.g., telnet or ftp) other than terminal connections between the two networks.
3. The TGW will not support connections from devices other than a TS.

TS-TGW Communication Protocol

The TS and TGW exchange information between themselves using a non-standard network layer protocol on top of Ethernet’s link layer service (see Figures 3 and 4). It is the purpose of this section to describe the packet format used and the function of each of its parts.

At the data link layer the packets follow the standard Ethernet packet definition. The only feature unique to this application is the value chosen for the Link Service Access Point (LSAP)—the TS uses a value of two in this field. While this value has a specific meaning in this context, it may have other meanings outside the EE/CISnet environment.

All user data and control information exchanged between the TS and TGW at the network layer have two components. The first part is the line number and the second part is the data destined for that line. If the high bit of the first byte containing the line number is set, this flags a control message for that line and the second byte is then interpreted as a control code for that line. Control messages include setting up and terminating connections, end-to-end flow control, status messages, autologout requests and TS-TGW Ethernet flow
control. Within a given packet, control and data pairs for the same or different lines may be multiplexed. Packets are disassembled two bytes at a time using the line number to direct each character to the correct destination. Between the TS and the TGW a lock-step flow control is observed. This prevents the TGW from ever overrunning the TS.

**TGW Operation**

The operation of the TGW can be best understood by first referring to Figure 2. When a user types a character at a terminal, that character is sent over serial lines to the EE/CIS terminal switch; the TS packetizes the character and sends it to the TGW. When the TGW receives the packet it disassembles the packet and forwards the character over the appropriate serial line to the MUX which sends the character to the remote MUX where it is then received by the ACS port selector and forwarded to the appropriate host. The character echo follows the same path, only in reverse order.

Figure 5 illustrates the internal operation of the TGW. The TGW buffers all data, both incoming and outgoing. When a message is received from the Ethernet, the message is stored in a buffer and the TGW begins to extract line number/character pairs from that buffer. Those pairs that represent normal data are queued to the appropriate serial output buffer based on the line number present in the pair. For the data representing control messages, the effect can be in either of two directions: if the control message is requesting that a connection be established, then a message is forwarded back to the requester by queuing an appropriate message to the current Ethernet output buffer; if the control message is requesting flow control for a particular serial line, then that request is queued to the appropriate serial line buffer.

When a character is received over one of the serial lines, an interrupt signal is generated and as soon as the TGW processor is available the character is processed. Initially, the character is placed on an input queue associated with the particular serial line. The character handler removes the character from that queue, prepends a line number, then appends the line number/character pair to the current outgoing Ethernet buffer.

Much of the software is interrupt driven; the device handlers all react to incoming data in this manner. The main program is an infinite loop that periodically polls devices and handles the flow of data in both directions. Packets are sent from the TGW over the Ethernet either as a result of filling with data or as a result of the current polling interval expiring. Because the PDP does not have a hardware clock built in, one was implemented in software based on interrupts supplied by a DL11-W board installed in the PDP. This clock is used to control device polling and to schedule events that are time based, such as the timing of events required to disconnect a line from the Port Selector. The clock also schedules the sending of statistics on a periodic basis when that option is configured into the software.

**Hardware Configuration**

The terminal gateway software executes on a dedicated DEC PDP-11/34 minicomputer. The code development was done on a VAX 11/780, cross-compiled for the PDP and downloaded via the Ethernet. The PDP is configured with 128K of main memory, a 3Com 3C300 Unibus Ethernet Controller, 1—DZ-11A Asynchronous 8-line Multiplexer, 2—DH-11A Asynchronous 16-line Multiplexers, and a DL-11W Serial Line Unit/Real-time Clock Option.

Communication with the remote network is through a pair of Tellabs 330C Dataplexers, a statistical multiplexer with 32 asynchronous channels. The Dataplexers communicate over public leased copper circuits using a pair of Amdahl 982 synchronous data sets which communicate at 64 Kbit/s. The port selector, with which the multiplexer interfaces at the ACSnet, is a Develcon Datasync (Model 9006/2000). As presently configured, the Datasync supports approximately 1440 channels or 720 user connections to ASCnet hosts.

The terminal switch software executes on a dedicated PDP-11/45 minicomputer, configured with 192K of memory, a 3Com 3C300 Unibus Ethernet Controller, 7—DH-11A

![Figure 5 — Block diagram of the TGW software](From the collection of the Computer History Museum (www.computerhistory.org))
Asynchronous 16-line Multiplexers or their equivalent, and a
DL-11W Serial Line Unit/Real-time Clock Option. The TS
provides for 112 terminal connections to the EE/CIS network
(both hard wired and dial-in) and is the principal connect
point to the EE/CISnet systems for all EE/CIS faculty and
graduate student terminals.

TGW PERFORMANCE METRICS

The TGW has been designed with an option to maintain cer­
tain statistics on itself. This was done to monitor the TGW’s
performance over time and to supplement statistics already
gathered by an existing network measurement center.\(^1\) A sample
of the counts kept by the TGW is summarized in Table IV. Certain special events are also counted: the number of con­
nections refused by the TGW and the number of connections
automatically logged out, provided that option has been se­
lected within the TGW software. These counts accumulate for
a period of one minute and are then reset to zero. Before they
are reset the TGW sends a statistics packet to one of the Vax
hosts via the Ethernet where the packet is stored for later
processing. A daemon process running on the Vax captures
each packet and appends it to an individual file. A separate
file is created for each hour during which statistics are kept.

From these counts a program running on the Vax can derive
a variety of performance metrics. These metrics can be based
on different intervals of time depending on how the data on
file are interpreted and accumulated. Several possible metrics
are listed below along with an example of how each metric is
computed.

Channel utilization of the 64 Kbit/s link: This metric is a
percentage based on the maximum capacity of the channel
being measured.

\[
\frac{(\text{Total Char Sent})}{(\text{Max. Possible Char Sent})} \times 100 = \% \text{ Utilization} \quad (1)
\]

A histogram of the average channel utilization for each
minute of the day over some period of time provides a com­
 pact way to characterize the level of use of the gateway.
Mean number of lines in use: This metric is an absolute
quantity and must be measured over a period of time.

\[
\text{Line}_i = \text{number of lines active during the } i\text{th minute} \\
N = \text{number of minutes processed}
\]

\[
\text{Mean } = \frac{\sum_{i=1}^{N} \text{Line}_i}{N} \quad (2)
\]

Comparing a histogram of the mean number of lines in use
during each minute of the day with the channel utilization
histogram suggested above would estimate the relation be­
tween number of users and the 64 Kbit/s channel utilization.
Presenting this metric as a histogram showing the mean num­
ber of lines in use for each hour of the day, as summarized
over several weeks of observation, would provide an indica­
tion of typical patterns of use of the TGW resource. Such a
histogram would be computed in the same manner as is sug­
gested for the mean number of connections refused, which is
presented below.

Correlate the mean number of users to the internet channel
utilization: One might expect to find a linear correlation be­
tween these two metrics. At some point, however, as the
number of users increases, the internet channel utilization
must level out (because it will be approaching 100%). If the
channel utilization turns out too high when the number of users
is significantly less than the maximum, this would be an
indication that the service provided is probably less than sats.
factory and that steps should be taken to increase the channel
bandwidth. One step might be to assign 16 lines to each of two
64 Kbit/s links instead of 32 lines to one link.

Mean transmission (or reception) rate for a connection: This
rate must be expressed as a unit quantity per unit time (e.g.,
characters per second).

\[
\text{Char}_i = \text{number of chars transmitted in the } i\text{th minute} \\
N = \text{number of minutes processed}
\]

\[
\text{Char/sec} = \frac{\sum_{i=1}^{N} \text{Char}_i}{60 \cdot N} \quad (3)
\]

This rate can be compared to the possible data rate for a line
(9600 baud) and if this comparison is done, it can be expressed
as a line utilization rate (%). This metric provides a compact
way to characterize the typical load that a single user presents
to the system.

Mean packet arrival rate: Since the number of Ethernet
packets received, or sent, is counted each minute, it is possible
to compute the mean number of packets arriving per minute
and to approximate the mean number of packets arriving per
second. For example, to compute the number of packets per
second over \(N\) minutes of observation the following formula
should be used:

\[
\text{Packets}_i = \text{number of packets arriving in the } i\text{th minute} \\
N = \text{number of minutes processed}
\]

\[
\text{Packets/sec} = \frac{\sum_{i=1}^{N} \text{Packets}_i}{60 \cdot N} \quad (4)
\]

It should be understood that this metric does not reflect an
accurate measure of packet interarrival time. This metric does
offer one measure of the amount of Ethernet bandwidth con­
sumed by the TGW-TS traffic.

Mean number of connections refused: This metric would be
interesting to view as a histogram showing the mean for each
hour of the day as summarized over several weeks of observ­
ation.

\[
\text{Mean (hr)} = \frac{\sum_{i=1}^{\text{Numdays}} \sum_{k=1}^{60} \text{# refused in min}_k \text{ of hr}_i}{\text{Numdays}} \quad (5)
\]

A display of the total number of connections refused each
hour of the day over an extended period compared with the
next metric would provide a means of estimating whether the TGW resource is being consumed by idle connections or by users actively working. This metric also gives some indication of user satisfaction with the TGW. If a large number of connections are refused on a regular basis, then the service provided has to be expanded or the period of time before idle users are logged out needs to be shortened.

Mean number of users automatically logged out per hour:

\[
\text{Mean} (hr.) = \frac{\sum_{i=1}^{\text{Numdays}} \sum_{k=1}^{\text{60}} \# \text{ automatic logouts in min}_k \text{ of hr}_i}{\text{Numdays}}
\]  

(6)

As with the previous measure, it would be interesting to have a histogram summarizing this metric for each hour of the day. This metric may provide some indication of the manner in which people use their terminals. Large numbers of automatic logouts would indicate a sporadic, intermittent kind of use.

Overhead on the Ethernet: Because each data packet sent and received over the Ethernet is a fixed size and is also categorized according to the number of data bytes, it is possible to make a rough approximation of the amount of Ethernet bandwidth wasted due to partially filled data packets. The measure is an approximation because the granularity of the data byte count kept as a statistic is only accurate to the nearest hundred bytes. For an individual packet the amount of overhead in bytes is given by:

\[
\text{Overhead} = \text{Packet Size} - \text{Fixed Overhead} - \text{Data Bytes Sent}
\]

where

\[
\text{Fixed Overhead} = \text{Preamble} + \text{Source Address} + \text{Destination Address} + \text{SAP} + \text{FCS}
\]

= 26 bytes (See Figure 3 for sizes.)

The Fixed Overhead is subtracted because it should not be counted as part of the wasted bandwidth. It is required that every packet have these components, therefore they are just as important as the data. For a given hour the estimated mean overhead would be computed as:

\[
B_i = \text{Number of data bytes sent in the } i\text{th minute}
\]

\[
P_i = \text{Number of packets sent in the } i\text{th minute}
\]

\[
\text{Mean Overhead} = \frac{\sum_{i=1}^{\text{60}} (P_i(\text{Packet Size} - 26)) - B_i}{P_i} \text{ bytes/packet}
\]  

(7)

Analysis of the overhead could be useful in determining an optimal size for the packets used by the TGW. If the overhead is large and the mean number of users is near the maximum, then the packet size should be able to be reduced.

CONCLUSIONS/SUMMARY

Due to the increasing costs of leased communications facilities there was a need to consider an alternative means of providing terminal communications to geographically separate LANs. This search for alternatives led to the proposal that a TGW be built. The cost of communications affected by the TGW has been analyzed and the benefits of using the TGW have been demonstrated. The terminal gateway provides several advantages over other possible communication schemes: per line costs are significantly reduced, a standard user interface is provided, inactive terminals may be automatically disconnected, malfunctioning lines may be isolated without disrupting other users, and more users can be serviced than with an equivalent number of dedicated lines.

The TGW provides a path between the two networks that off-loads a host or dedicated general purpose gateway on each network from the tasks associated with providing an interactive communications channel; thus the TGW allows a general purpose gateway to handle its other tasks of file transfer and message forwarding more efficiently. The TGW presents some disadvantages over dedicated lines: individual line performance is degraded in the face of heavy use, failure of the 64 Kbit/s link or its associated hardware is catastrophic for all gateway users (although users may still attach to the remote hosts using login or telnet from the EE/CISnet hosts), and the cost to add lines beyond the capacity of the MUX is a large, one-time expenditure.

The TGW has been described in terms of the ISO-OSI reference model and has been characterized as a special purpose gateway. The design and operation of the TGW software and hardware necessary to make the system functional have been described. The special nature of the traffic handled by the TGW made it possible to remove some of the functionality normally found in a general purpose gateway. Tasks like routing, address verification, fragmentation, and reassembly are not required. Because of this the TGW can more efficiently handle the data transfer assigned to it, allowing satisfactory real-time performance to be achieved. The requirements and restrictions placed upon the TGW have been presented.

Several performance metrics are available for measuring the performance of the TGW. A mechanism for gathering the counts necessary for computing these metrics has been implemented. It is possible to obtain significant statistical results from the data maintained by the TGW; such an analysis could yield a characterization of the gateway usage and a picture of the requirements of the clients served by the TGW.

The TGW has become a functional part of the computer communications facility of the EE/CIS computer lab and has been found to be a cost effective alternative to leasing communications services. As the University expands its networking capacity, other alternatives will become available for interconnecting existing LANs. However, the TGW will continue to serve a useful role in providing users on a local LAN with a uniform mechanism for terminal connections to hosts on a remote LAN. Until such time as the TS networking software is ported to the other types of machines available on the ACSnet (e.g., the IBM 3081), a general purpose gateway will
not be sufficient for allowing the TS to communicate with those hosts. Because the TGW provides a transparent, asynchronous, serial communication channel (RS-232) for the terminal attached to it, the TGW does not depend on any proprietary protocols. Therefore, the TGW can provide connections to any device that supports RS-232 connections. In our case, this is the ACSnet Port Selector.

The TS provides a cost-effective way to allow users to make multiple virtual connections to machines on a LAN to which the TS is attached, and the TGW extends the TS's capacity by allowing connections to devices on another local network to occur in a manner that makes it appear as if the device were just another node on the local network. This connection is accomplished using leased communication facilities and represents both a technically feasible and economical solution which could be implemented by other users with similar requirements.

REFERENCES

The growth of technology has increased at a rapid rate in recent years. Undoubtedly, advancements in technology will continue to increase and change society as the United States moves into the Information Age. As with other areas, high technology has shed its effects on society before policymakers could come to grips with its societal implications.

Is the Nation able to move into the Information Age and retain its beliefs in the principles of the U.S. Constitution? The opening session focuses on the increasing capability to disseminate information rapidly and efficiently and its impact on an individual’s right to privacy. The intent and accomplishments of the Privacy Act of 1974 along with current developments in the area of information privacy are assessed, and the argument that information privacy is eroding as a result of new technologies is addressed.

Is the Nation able to protect its citizens and businesses from new types of crimes derived from advancements in technology? Another area where the law has not kept up with the technology is in the Nation’s criminal codes. Three sessions in the Security, Privacy and Law track will address computer crime legislation, law enforcement and prosecution, and corporate security measures.

The U.S. Congress recently adopted the Computer Fraud and Abuse Act of 1986. Most states have passed computer crime legislation. However, the laws vary in their ability to effectively prosecute those involved in computer crime. Federal and state laws regarding computer crime are reviewed to identify what constitutes a computer crime.

Security measures adopted by corporations have proved to be the best way to defend against computer crime. Security specialists explain how an organization can address the task of developing or assessing a program to protect computers and information systems.

If a breach of security or a computer crime is suspected, what procedures should be implemented to investigate the matter and make a case? How should the matter be disposed of? Law enforcement agencies have only recently addressed the need for specialists in tracking down computer criminals. Experts in computer crime investigations describe the investigating team that should be assembled, the relationship between governmental and private police work, the rights of suspects, the identification of evidence, and prosecution policies.

Software transactions and contracting for services has raised questions not easily identifiable in current law. One session focuses on the relationship between vendors and users—their respective needs and obligations to each other. The session will explore the legal and practical aspects of contracting for computer goods or services. Discussion covers protection and use of software, liability for delivery or systems failures, financing and payment, and tax implications.

Is the Nation capable of maintaining its leadership position in the world economy? Can the workforce be prepared for new job requirements as the Nation moves from a manufacturing-oriented economy to a service-oriented economy? Two sessions in the track address the overall issues revolving around high technology and public policy.
One session discusses options for the development and implementation of a national information policy. Questions raised involve the relationship among Federal, state and local governments in policy development; appropriate government mechanisms for policy oversight; and the extent to which the private sector should be free from information regulation.

The final session explores the role, if any, of Federal and state governments in encouraging the economic development of computer technology and research. The international implications of market share and protection, technology transfer, intellectual property protection, investment tax benefits, and antitrust implications are also discussed.

The Information Age and the growth of technology affect every segment of society. The Security, Privacy and Law track provides a better idea of the issues and concerns that impact all of us.