Excerpt from BBN Report 3056

The Atlantic Satellite Packet Broadcast and Gateway Experiments

by

R. Binder
R. Rettberg
D. Walden

2. THE ATLANTIC SATELLITE GATEWAY EXPERIMENT

The work done to date on the interconnection of computer networks has usually assumed a configuration such as that shown

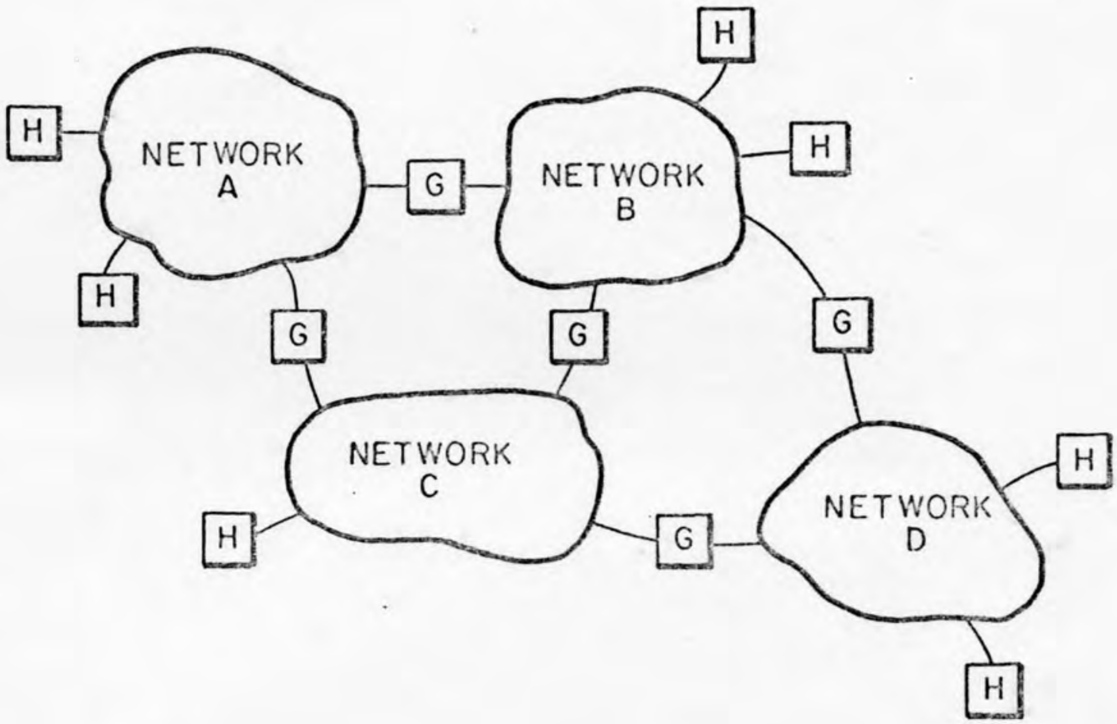


Figure 1 -- Networks Connected by Gateways

in figure 1. On each network there are hosts (denoted by H in the figure) which desire to communicate with hosts on other networks. The networks are connected together by units (denoted by G) called "gateways" (note this is a different use of the term "gateway" than in conventional international telephone system interconnection, where the term is used to refer to an artificial site from which it is convenient to establish tariffs). The gateways must in some way convert traffic in the format of one network into traffic in the format of another network.

Because host-to-host protocols differ from one network to the next, and because these protocols are generally complicated and incompatible, it seems clear that hosts on different networks wishing to communicate must do so in a common protocol. Much of the work to date in network interconnection has been to specify such a standard protocol: see for example [Cerf 74a], [Zimmerman 74], [McKenzie 74]. While there has not yet been final agreement on the standard protocol, we suggest that the network interconnection experiment use the protocol described by Cerf and Kahn [Cerf 74a] as a starting point. In this protocol the logical entity in the host which performs the protocol functions is called the Transmission Control Program or TCP. In later portions of this proposal, we will refer to the TCP and its protocol, with the realization that its use has not yet been accepted for this experiment or as an internetwork standard.

While there has been considerable work on the standard host protocol, to date there has been less work on the function and structure of the gateway. For the most part it has been assumed that the gateway will forward traffic between networks (and across networks) without specifying how this would be done. The purpose of the experiment we propose will be to investigate the functions the gateway should perform and how it should perform them. (See [Burchfiel 74] and [Lloyd 75] for related experiment proposals.)

2.1 Gateways as Hosts vs. Gateways as Nodes

One of the outstanding questions of network interconnection is whether the gateways should connect networks at the packet or host level. By packet level, we mean that a portion of the gateway would actually become a node on each of the networks being connected. By host level, we mean that a portion of the gateway would actually be a host on each of the networks being connected. We feel that the gateways should connect at the host level primarily to maintain the sovereignty of the networks involved [Crowther 72]. Furthermore, it is unlikely that a standard packet format can be found, at the present state of ongoing development of all of the packet-switching networks that might be connected, which would permit packet level connection.

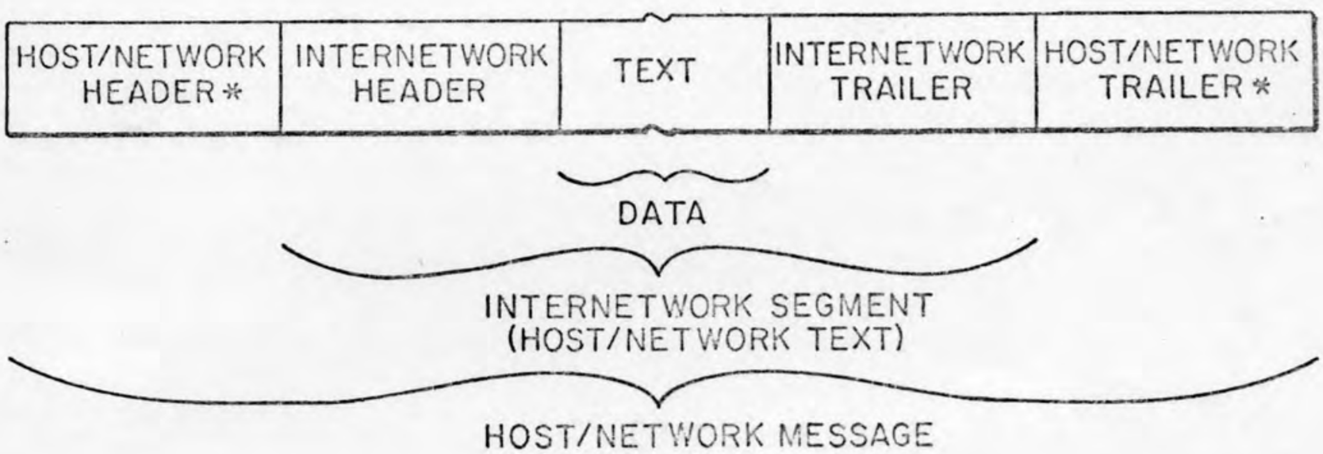
By network sovereignty, we mean that connection to the networks must be done at a point where the interface is both well defined and well controlled by the constituent networks. If the point of connection is the host level just mentioned, each network can protect itself against activities of the gateway to the same extent as it may protect itself against the activities of any other host.

As already mentioned, it seems clear that it will be impossible, in general, for the gateway to convert between the host to host protocols of two communicating networks. Thus, rather than communicating in the host to host protocol of the network, the gateways should communicate with nodes of the network in the lowest form of host/network protocol supported by the network. Transmissions in the network interconnection protocol of the TCPs should be the text of these host/network protocol messages.

Notice that a host on a given network might find itself having to implement, in addition to the host to host protocol of its own network, the standard internetworking host to host protocol for communication with hosts in other networks. Of course, one can hope the internetwork standard will eventually prevail throughout the world and the host to host protocols of the individual networks will eventually wither away.

To summarize, hosts involved in internetwork communications must adopt a common protocol, and gateways should connect to networks as hosts using the lowest level of host/network protocol. Further, a protocol has already been specified [Cerf 74a] for the former task and the protocol for the latter task is already specified by the network for any network to which a gateway is to be attached [e.g., BBN 74].

If the gateways connect to the networks as hosts, then the format of the messages passed to the network are specified by the host/network protocol. This protocol is then used to permit transparent transmission of segments of an internetwork transmission. This can be done by embedding the internetwork segment in the text of a message in the host/network protocol as



*THIS HEADER VARIES FROM NETWORK TO NETWORK

Figure 2 -- An Internetwork Segment Embedded in a Network Message

shown in figure 2. Such a composite message has two leaders and potentially two trailers. The outermost leader and trailer provide information for the network. The leader will specify the address of the gateway host to which the message should be delivered, any allocation or sequencing information which is used by the host/network protocol, and any further information demanded by that protocol. An example of a trailer that might be required by the host/network protocol would be padding and a checksum. Within this outermost leader and trailer is the internetwork data segment with its leader and trailer. The internetwork leader specifies such information as the ultimate destination, sequencing, and reassembly information. The actual data which is being transferred is the text of this message.

We will now examine some of the characteristics a gateway must have in addition to being able to pass messages between two networks.

2.2 Gateway Characteristics

The gateway must have the following additional characteristics:

- -a capability for inter-gateway routing
- -access control and accounting mechanisms
- -control of congestion at the gateways
- -fragmentation
- -in some cases, a capability for inter-gateway retransmission

In the following paragraphs we elaborate on each of these points.

must have the ability to fragment a larger message arriving from one network into smaller messages which are acceptable by the next network. When such fragmentation occurs, the message stream must eventually be reassembled into its original structure. The protocol proposed in [Cerf 74a] provides this function at the destination host.

<u>Congestion Control.</u> Congestion will inevitably occur at the gateway unless specific measures are taken to prevent it. This congestion can occur as a result of speed mismatches between the networks connected by a gateway, because several gateways on a network may simultaneously transmit traffic to the same other gateway, because traffic may have to be held during a period of recovery from a failure, and so on. One specific kind of congestion results from deadlocks, such as when gateway A is full of traffic for gateway B which is full of traffic for gateway A.

<u>Retransmission.</u> When a message is lost in the network between two gateways, one can either retransmit the message between the two gateways or assume that the message will be retransmitted from the source host to the destination host. It has been shown [NAC 73] that hop-to-hop retransmission is more efficient than source-to-destination retransmission if the possibility of message loss is appreciable; and even when there is little possibility of message loss, the variance of retransmission delays is less with hop-to-hop retransmission than

with source-to-destination retransmission. While some networks deliver messages very reliably, other networks rely on source to destination retransmission and in some cases are quite cavalier about throwing away messages. Thus, the gateways should have the ability to retransmit messages across lossy networks. It seems that at least when the hosts on a network are normally responsible for retransmission across that network, the gateways ought to provide retransmission across that network. ([Mader 74] supports the notion that end-to-end timeout and retransmission can be unduly inefficient.)

2.3 The Gateway Virtual Network

Notice that the characteristics of a gateway described above are very similar to the characteristics of a node on a packet switching network [Crowther 75]. This leads one to the notion of a gateway virtual network wherein the gateways act as nodes and the network spanned by the gateways acts as virtual lines fully connecting all the gateways on that network. This concept is illustrated in figure 3, where network C has been replaced by the virtual lines it provides. Further, notice in the figure that logically there is a gateway associated with each host attempting internetwork communication.

Although the figure shows a separate gateway for each host, we do not mean to imply the gateways must be physically separate

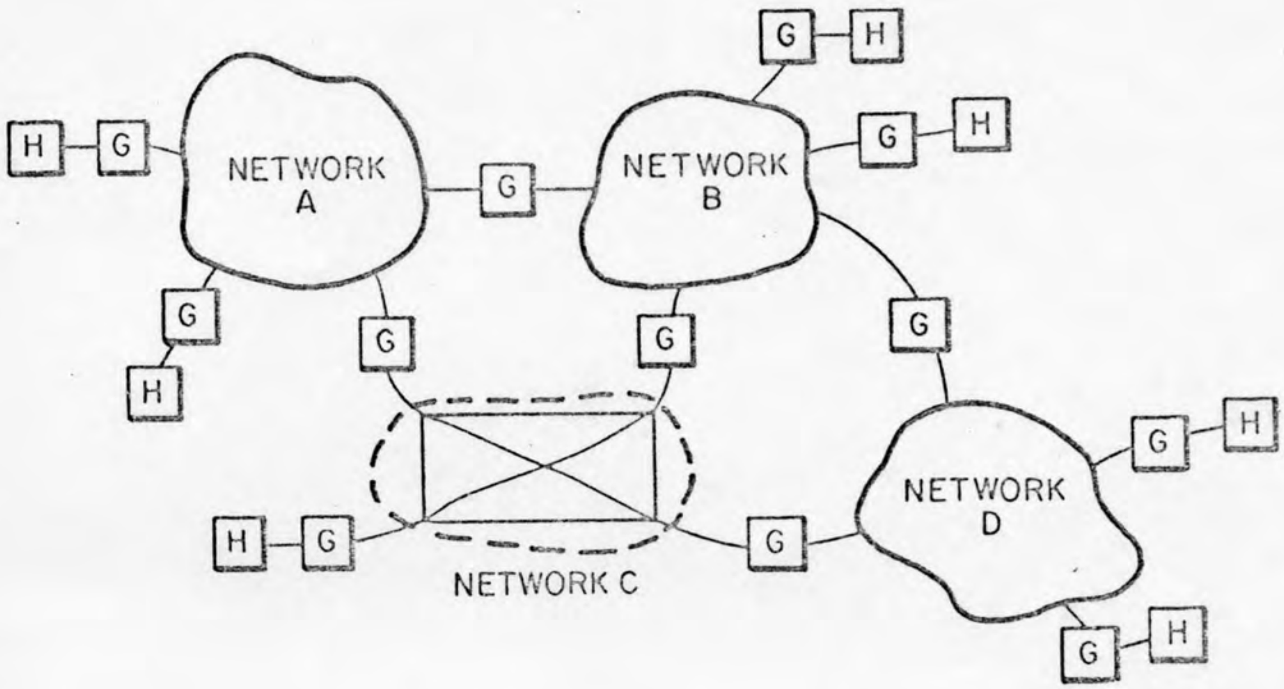


Figure 3 -- Gateway Virtual Network

machines or that there need necessarily be a one-to-one correspondence between hosts and gateways. For instance, the logical entity that is the gateway may have the form of a program running in a host computer. Alternately, the gateway could be in a stand-alone machine connecting two networks or serving one or more hosts. On the other hand the gateway connecting two networks could even take the form of a program running in a host which is connected to both networks. In general, a gateway should be able to connect any combination of hosts and networks. (The political and economic issues relating to whether the gateway should reside in a host or a separate gateway machine are discussed in [Kuo 75].) Furthermore, it is entirely possible for a host not to have its own gateway at all, preferring to use a gateway elsewhere in its network to perform the gateway functions for the host. In this case, the host would simply know the address of a couple of gateways in its network and would send its internetwork traffic arbitrarily to one of these gateways for routing and forwarding. Note that the gateway arbitrarily chosen may not be on the best path to the destination, and this gateway may, in fact, forward the traffic to another (better) gateway in the same network for forwarding outside the network. Thus, we see that this approach may be inefficient but it reduces the number of gateways that have to be constructed. Because it reduces effort, we will carefully consider use of this latter approach for our experiment. Further, we believe both approaches should be supported in the long term; since the two approaches are compatible, this causes no problems.

It is our view that the gateway network should have many of the maintenance characteristics of a stand-alone packet switching network [McKenzie 72]. It should have centralized development and maintenance responsibility, including a gateway network monitoring and control center (GCC). We do not, however, feel that the gateway program must run in any one brand of machine; indeed, it will probably be necessary to support the gateway program on machines of several nationalities because of the international extent of these networks. This may make it attractive to specify (and if possible provide) the program in a common, universally supported programming language.

The Gateway Virtual Network solution is a general solution to the problem of interconnecting networks which is not highly dependent on the nature of the networks being connected. Because of this we expect that networks which are connected in the future will not require modification. While one can conceive that a simpler but less general form of interconnection may be possible through modification of the current networks, future networks may be much less amenable to this form of attachment and therefore require major modifications to themselves or even to the previously interconnected networks.

2.4 Division of Functions Between the Gateway and TCP.

Just as in a stand-alone network there is a question of the division of responsibility between the hosts and the nodes (i.e., should reassembly be done by the hosts or should the nodes deliver traffic in order), in the gateway virtual network there is a question of the division of responsibility between the TCP and the gateway. It is best to implement these functions in either the gateway or the TCP, rather than blurring the implementation across the boundary between them. Some points are quite clear. For instance, reassembly must be done by the destination TCP, as traffic traversing the virtual network may be fragmented and pieces routed on alternate paths to the destination. Thus, the ultimate destination TCP must be prepared to rearrange the communication stream into the correct order and reassemble it into the internetwork data segment. (The reassembly problem is further aggravated when intervening networks do not maintain ordering.)

Second, the differences in message or packet size between constituent networks mean that it may be necessary for messages to be successively fragmented into smaller and smaller units as they pass from one network to the next. It seems natural that this task should be done in the gateway since it knows the message and packet size characteristics of the networks to which it is connected. Even at the source host, the TCP can leave any necessary fragmentation of the message stream to the gateway.

Just as clearly, the gateways must be responsible for the routing calculation, since only the gateways have the global knowledge necessary to make a sensible routing decision.

The access control and accounting functions should also be in the gateway since these functions are desired between the networks.

It is less clear whether the gateways or TCPs should perform the congestion control and retransmission functions. One alternative would be for the gateways not to worry about retransmission and to solve any congestion problems simply by discarding traffic, in each case relying on the source and destination TCPs to provide the necessary recovery mechanisms. In fact, the TCP of [Cerf 74a] does provide such recovery mechanisms. On the other hand, if much traffic is discarded by the gateways to control congestion or if there is even one network which loses traffic frequently, then we believe relying on source to destination TCP retransmission will be prohibitively inefficient and will also be expensive, both in direct (network-imposed accounting) and indirect (TCP overhead) costs. For this reason we think it is incumbent on the gateways to shoulder the burden of controlling their own congestion and for the gateways to provide the option of retransmission across a lossy network. Of course, since the source to destination TCP mechanisms will exist, there is perhaps no need for the gateways to perform these functions at the outset of the experiment, these functions being added somewhere downline. If the initial gateway implementation provides congestion control by discarding segments, we will take statistics on the frequency of discard. In addition, even after the gateways provide these functions, the TCPs should retain the end-to-end retransmission capability at their level for reliability, since retransmissions performed by the gateway level are for efficiency rather than complete reliability.

The remainder of the functions performed by the [Cerf 74a] TCP are properly the functions of the TCP and not the gateways, as these functions are concerned with end-to-end issues, user process level issues, etc., while the gateways are properly concerned only with traffic switching issues.

There does have to be some communication between the TCP level and the gateway level. Most obviously, the TCP must specify the address of the destination TCP to which the gateways are to route the traffic. This particular communication can be effected simply by having the gateways understand the TCP traffic formats which include such addressing information. Another area of communication required between the TCPs and gateways is to specify certain transmission characteristics for the traffic (e.g., networks through which the traffic must not be routed, maximum acceptable delays, and average throughput required over a period of time). This area has not been explicitly addressed previously and requires further study.

At present we have no opinion on whether the security function should go in the gateway or the TCP. Current COMSEC standards appear to require pairwise source host to destination host encryption. If such standards are maintained, then the security function must obviously reside with the TCPs. Alternatively, there has been much discussion of so-called "link encryption." Use of link encryption would lead one to place the security function in the gateways. A third alternative would be to place the security function between the source and destination TCPs and their gateways. In this case the TCP "header" must be in the clear, permitting the gateways to access the information. There are elaborations of all the above schemes and other schemes are possible. We mention the security issue here merely for completeness; we do not intend to provide this function for the gateway experiment.

2.5 The Phasing of the Gateway Implementation

As stated in the preface, the Satellite IMPs will be delivered to Goonhilly and Etam without any gateway code. They will join the network initially as standard IMPs with special modem software for the satellite links. This software is essentially complete. For an initial period only experiments on the satellite channel will be performed. This mode of dedicated operation may be resumed at later stages in the project as necessary and under the control of the NCC.

After the initial period we plan to introduce gateway software into the Satellite IMPs. Once this is done, the satellite portions of the Satellite IMPs will become a network independent of the rest of the ARPA Network, linked to it by the gateways. At this point, normal ARPA Network host level communications with the Satellite IMPs will be impossible: all communications with or through the Satellite IMPs must be at an internetwork (or TCP) level. The gateway code will be placed in the Satellite IMP logically between the line to the ARPA Network and the IMP portion of the Satellite IMP as illustrated in figure

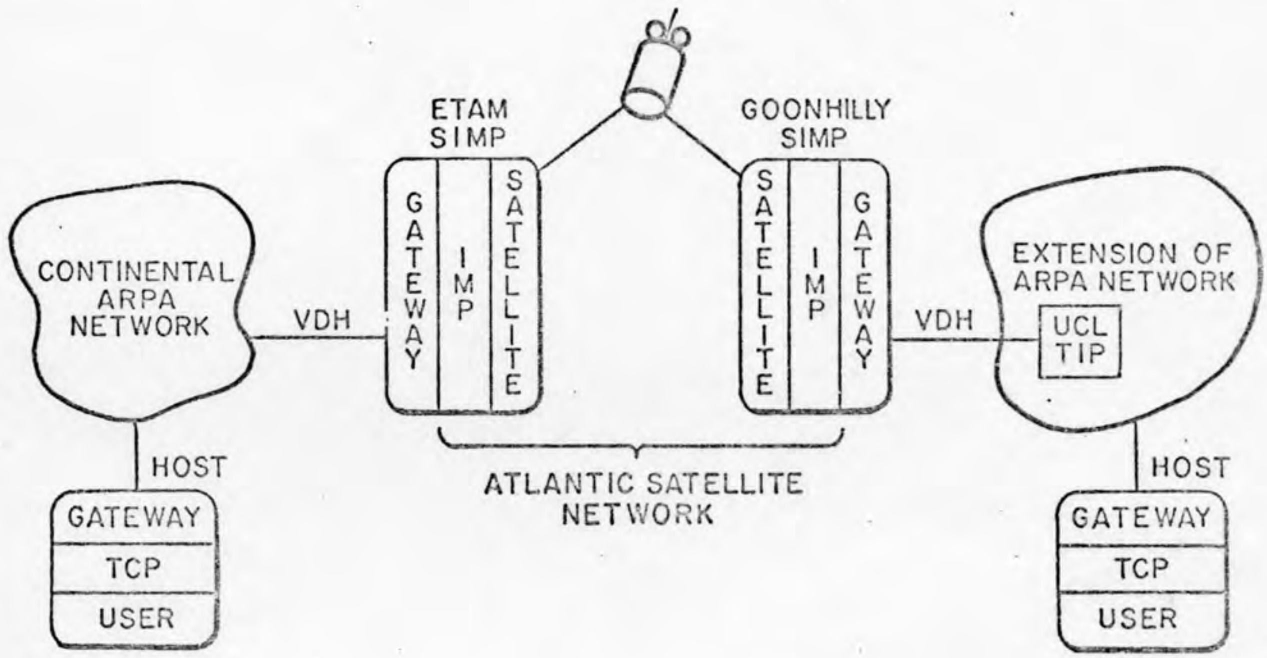


Figure 4 -- The Atlantic Gateway Experiment Configuration

4. The gateway code will connect to the ARPA Network as a Very Distant Host and to the IMP in the same machine over a simulated host interface. The interconnection between the IMP and satellite handling portions of the Satellite IMP will be the same as before the inclusion of the gateway code.

The gateway will, at this point, provide the functions of packet fragmentation, routing, interfacing to the connected networks, and such "access control" as is inherent in the fact that the gateway can only be accessed as a host on the networks. At some point later in this program, we expect to adapt the gateways to be supported independently of the IMP, complete with restart, reset, and statistics capabilities of their own. At this point, we will also be prepared to provide access controls as agreed upon with ARPA.

2.6 The Details of the Initial Implementation

The implementation of the gateway is illustrated below. It consists of modular structures which carry on communications at a message level (VDH or Local Host) with various networks, modules which transform a particular network message format into a TCP type packet and vice versa, and modules which perform message routing and other centralized message processing functions such as flow and access control.

The purpose of each of these elements of the gateway will be clear if we follow a message as it is processed by the gateway from Network 1 to Network 2 in figure 5. The line to Network 1 is operated by the code labelled "N1 Host". This code acts as a host on that network, passing the required control information and transferring messages to and from other hosts on that network according to the lowest level message protocols of that network.

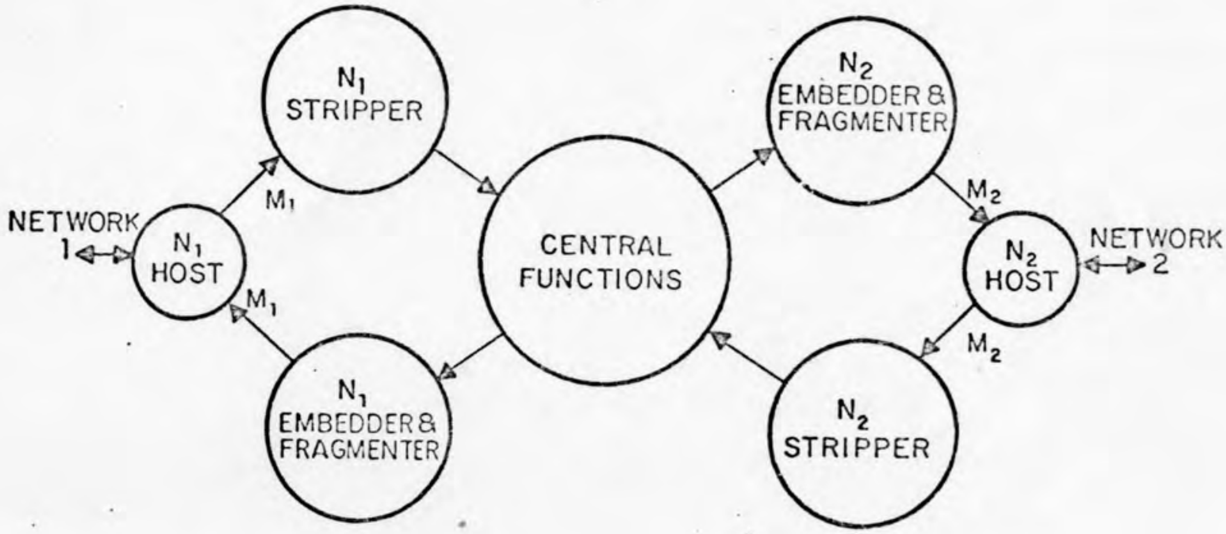


Figure 5 -- Components of a Gateway

In the case of the connection to the ARPA Network, this code will implement the standard Very Distant Host protocol. The atomic unit of data transfer for this code is the message of the ARPA Network which will allow testing with three different traffic types: multi-packet messages, single-packet messages, and "uncontrolled packets" [BBN 74]. Incoming messages received from Network 1 are passed to the "Network 1 Stripper," where the

Network 1 message leader is removed. The result is an internetwork protocol segment. This segment is passed to the central portion of the gateway, which examines the address specified in the segment and queues it for output to that network. The "Network 2 Embedder and Fragmenter" takes these internetwork segments and converts them into messages in the Network 2 format. In this case, once again, the format will be ARPA Network messages. If the internetwork segment does not fit within the text of a network message, it will be fragmented and re-formatted until all parts fit into network messages. (In our implementation, we will include a parameter which will specify the maximum message length so that this portion of the program may be fully tested.) Finally, the message is presented to the Network 2 message interface (in this case a software host interface to the IMP in the Satellite IMP) for delivery to the appropriate gateway host.

The control function will be responsible both for determining where to route an individual internetwork segment and for performing the distributed routing functions in cooperation with the other gateway nodes. In operation, this distributed routing will be accomplished in the following way. Each gateway will exchange routing packets, containing information about all gateways in all networks, with every other gateway on each network to which it is attached. The gateway will be given the host addresses of all other gateways on each of the networks to which it is attached in an internal table which will be used eventually for the access control logic. This is similar to the way an ARPA Network IMP exchanges routing packets with its immediate neighbors, where the packets contain information about every possible destination; the other gateways on the same network are logically the immediate neighbors in the gateway network. In this case, however, it is necessary for each gateway to contain internally the network address of the other gateways in the same network, because while an ARPA Network IMP has at most five neighbors, a network has potentially many hundreds of hosts, a large fraction of which may be gateways.

The information passed in the routing messages can contain at least three types of measurement about the network: delay, bandwidth, and delay variability. In the initial implementation, we will accumulate delay and estimate bandwidth for the network connection between each pair of gateways in the same network. The gateways will then maintain a structure similar to that used by the IMP for routing. For each potential destination gateway, the intermediate gateway to which messages should be addressed for minimum delay and maximum bandwidth will be determined along with the expected delay and bandwidth. Periodically, the gateway will report to each of the other gateways on the same network the expected delay and bandwidth via this gateway. The other gateway will then add the delay to that gateway and will determine the minimum of the bandwidth through that gateway and the bandwidth on the path to that gateway, to determine the delay and bandwidth which may be expected along this path to the destination. Through this mechanism, each gateway will have the necessary information to route packets. Priority letters will be routed for delay, others will be routed for bandwidth.

We recognize that this routing scheme suffers from an inability to accurately determine the expected delay and bandwidth along a path, particularly as the actual path across a network may change unbeknownst to the gateways. This is intrinsic to the implementation of gateways as hosts rather than as nodes, since only the nodes of a network normally have the necessary state information (e.g., line speeds between node pairs or the number of packets queued for a line), and so far, no network passes this information to its hosts. We believe that the reasons mentioned previously in this note for connecting gateways at the host level are very important, and that the way to improve the routing efficiently between gateways is for the networks to pass expected delay and bandwidth information, for example, to their hosts. Even without the issue of gateway routing, it might be useful for hosts on a network to receive such delay and bandwidth information to improve their use of the network (e.g., deciding whether to send a large file now or later).

This eventual gateway implementation will provide many of the desirable features of an internetwork configuration.

However, in the interest of having an operational system available as soon as possible, we expect to put off several features in the initial implementation. Some of these desirable features may be provided by the IMP, since the first gateways will be in the same machines as IMPs. These features include 1) a gateway NCC to monitor the operation of the internetwork virtual network; 2) a DDT; 3) statistics; and 4) reloading of the gateway code. We initially expect that the use of the IMP's functions in these areas will be acceptable. In addition, we do not expect to implement message retransmission or acknowledgment across the network in the initial implementation. In general, the requirement for this is a function of the behavior of the associated network (some networks are liable to lose messages often). In this case, the associated network (the ARPA Network) has a very small chance of losing messages and the internetwork protocol can be relied upon to retransmit in the instance of message loss. Furthermore, we do not intend to include any explicit access controls or inter-gateway congestion control facilities in the initial implementation.

It is clear that a gateway will be required in association with each TCP. Since these gateways form a network of their own, routing and formatting packets, and since there may eventually be many of them, it is advisable to keep them as identical as possible. We therefore intend to specify the gateway routines in the only universally available language, FORTRAN. If possible, the gateway code in the Satellite IMPs will be machine compiled from that specification. If this is not possible, the routines will be hand compiled from the FORTRAN source. We hope that in this way, the implementation effort and variability will be minimized in implementations at the various hosts. Upon ARPA's request, we are prepared to extend this program to include a FORTRAN specified version of the TCP specified in [Cerf 74b] or as that specification is updated. We expect this implementation to require about four to six thousand words of code and about 2000 words of buffers. Since the satellite code and buffers require 5000 words, we expect an additional 8000 words (the minimum addition physically possible) of memory to be necessary and sufficient for the gateway routines for the purposes of this experiment. However, this will exhaust the machine's memory and for a production rather than experimental gateway it might be good to recode for a new machine with a larger memory address space, or to place the gateway code in a machine separate from the IMP and Satellite IMP.

2.7 Communication with Non-TCP Hosts

There are some special problems associated with this experiment. During the initial phase of the experiment, there will be no gateway; therefore, statistics and parameter change messages can be used just as in the regular ARPA Network. However, once the gateways are installed, any communications from the ARPA Network to the Satellite Network across the gateway host interface must be done at the TCP protocol level. However, the statistics and other fake hosts in the Satellite IMP are not capable of that protocol level. Furthermore, the internetwork protocol as specified in [CERF 74b] has no provision for packet trace. The latter of these difficulties can be avoided by tracing packets on entry to the Satellite Network. The former problem is, however, more difficult.

In general, the introduction of the internetwork protocol is not "backwards compatible". There is no way for a host with a TCP in one network to communicate with a host in another network which does not have a TCP. That is, there is no way to cut-through the high level protocol to deliver messages in the format of one of the constituent networks. We believe that it would be useful to provide a feature which permits cut-through from the TCP formats to the formats of a constituent network. Such a feature can provide a good solution to the problem encountered in transmitting statistics and parameter-change messages across the network boundary during our experiment and will be generally useful.

The simplest mechanism for connecting a TCP host with a non-TCP host is to use a series of sockets in a gateway on the same network as the non-TCP host. That gateway would itself contain a TCP and would therefore be able to communicate with any internetwork host. The gateway would maintain a series of host/network level links to the non-TCP host. Thus, a TCP host would communicate with a non-TCP host by opening a connection to one of these sockets. The TCP host would, then, send messages to the gateway with a special indicator set. In the gateway these messages would be reassembled, the internetwork leader removed, and the message transmitted to the indicated network non-TCP host as a new message. Any reply (e.g., RFNMs) to these messages would be packed within an internetwork message, and would be returned to the TCP host. The TCP host could receive other messages from the non-TCP host by way of the gateway TCP which is associated with this connection; individual conversations would be identified with a particular link.

REFERENCES

BBN 74.
Bolt Beranek and Newman Inc., "Specifications for the Interconnection of a Host and an IMP," BBN Report 1822, revised December 1974.

Burchfiel 74.
J. Burchfiel and R. Tomlinson, "An Experimental Simulation of a Satellite Gateway," INWG Experiment Note 2, August 1974.

Cerf 74a.
V.G. Cerf and R.E. Kahn, "A Protocol for Packet Network Intercommunication," IEEE Transactions on Communications, Vol. COM-22, 5, May 1974, pp. 637-648.

Cerf 74b.
V. Cerf, Y. Dalal, and C. Sunshine, "Specification of Internet Transmission Control Program," INWG Note 72, revised December 1974.

Crowther 72.
W.R. Crowther and D.C. Walden, "Response to INWG Note 6," INWG Note 10, December 1972.

Crowther 75.
W.R. Crowther, F.E. Heart, A.A. McKenzie, J.M. McQuillan, and D.C. Walden, "Issues in Packet-Switching Network Design," to be presented at the AFIPS 1975 National Computer Conference.

Kuo 75.
F.F. Kuo, "Political and Economic Issues for Internetwork Connections," Computer Communication Review, Vol. 5, No. 1, January 1975, pp. 32-34.

Lam 74.
S.S. Lam, "Packet-Switching in a Multi-Access Broadcast Channel with Application to Satellite Communication in a Computer Network," UCLA Technical Report, UCLA-ENG-7429, March 1974.

Lloyd 75.
D. Lloyd, M. Galland, and P. Kirstein, "Aim and Objectives of Internetwork Experiments," INWG Experiment Note 3, January 1975.

McKenzie 72.
A.A. McKenzie, B.P. Cosell, J.M. McQuillan, and M.J. Thrope, "The Network Control Center for the ARPA Network," Proceedings of the First International Conference on Computer Communications, October 1972.

McKenzie 75.
A.A. McKenzie, "Internetwork Host-to-Host Protocol," INWG Note 74, December 1974.

Mader 74.
E. Mader, W. Plummer, and R. Tomlinson, "A Protocol Experiment," INWG Experiment Note 1, August 1974.

Metcalfe 73.
R.M. Metcalfe, "Packet Communications," MIT Project MAC Report, MAC TR-114, December 1973.

NAC 73.
Network Analysis Corporation, "Comparison of Hop-by-Hop and End-to-End Acknowledgement Schemes," Packet Radio Temporary Note 7, January 1973.

Rettberg 73a.
R.D. Rettberg, "Slotting," ASS Note 42, May 1, 1973.

Rettberg 73b.
R.D. Rettberg, "Geometric Retransmission Randomization," ASS Note 51, September 1973.

Rettberg 74.
R. Rettberg and D. Walden, "A Proposed Experiment in Packet Broadcast Satellite Communication," BBN Report 2891, September 1974.

Zimmerman 74.
H. Zimmerman and M. Elie, "Transport Protocol. Standard Host-Host Protocol for Heterogeneous Computer Networks," INWG Note 61, April 1974.