

ISSUES IN INTERNETTING  
PART 1: MODELLING THE INTERNET

Eric C. Rosen

Bolt Beranek and Newman Inc.

May 1981

## ISSUES IN INTERNETTING

## PART 1: MODELLING THE INTERNET

## 1. Modelling the Internet

This is the first in a series of papers which attempt to deal with the problems of internetting in a comprehensive manner. By "internetting", we mean the establishment of data communications among some set of host computers which cannot all access the SAME data communications network (though we will, of course, require that each host be on some particular data communications network). The traditional approach to getting data from a host on one network to a host on another is to pass the data through an intermediate, or "gateway", host, which is a host on both networks. As we shall see, however, building an internet which is sufficiently robust, and which offers sufficiently high performance, to be useful in day-to-day data communications operations involves much more than simply pasting networks together in a pairwise manner. Rather, it requires us to build an entirely new network, which can be regarded as being hierarchically "above" the existent data communications networks. We shall see that building this new network is no simple task, but that it raises all the same issues that one must deal with in building any sort of data communications network. Our basic approach will be to consider SYSTEMATICALLY the ways in which an internet is and is not similar to "ordinary" data communications

networks, so that we can easily see how the knowledge we have gained from studying such networks (in particular, the ARPANET) can be applied to internetting. This paper will present a model for the internet which will help us to organize the issues; further papers will deal more explicitly with such issues as internet access, addressing, routing, and congestion control.

### 1.1 The Model Described

We begin by introducing a very general model for a class of abstract entities called NETWORK STRUCTURES. A Network Structure consists of three kinds of entities: SWITCHES, PATHWAYS, and HOSTS. When we say that a particular entity is a Host WITHIN SOME PARTICULAR NETWORK STRUCTURE, we mean that within that Network Structure it functions as a data sink and/or source, and does not perform such functions as store-and-forwarding traffic which originated elsewhere and is destined for elsewhere. By saying that a certain entity is a Switch WITHIN A PARTICULAR NETWORK STRUCTURE, we mean that, within that Network Structure, it is responsible for store-and-forward functions, i.e., for receiving data from a source Host, and sending it (possibly through a sequence of intermediate Switches) into a destination Host. A Pathway WITHIN A PARTICULAR NETWORK STRUCTURE is a communications path which has as one endpoint a Switch of the Network Structure, as its other endpoint either a Switch or a Host of that Network Structure, and NO intermediate Switches of that Network Structure.

It is important to understand that the notion of a Network Structure is intended to be an abstraction which we use in order to impose a conceptual organization on a set of physical entities. It makes no sense simply to ask of some particular computer, is it a Host or a Switch, unless one also references some particular Network Structure. Saying that a computer is, e.g., a Switch, attributes to it a certain functionality relative to a certain Network Structure. A particular computer might well be a Switch in one Network Structure while simultaneously being a Host within another Network Structure. (We will capitalize the terms "Host," "Switch," "Pathway," and "Network Structure" as a reminder of the abstract nature of these terms.)

Let's look at some examples. The ARPANET can be regarded as a Network Structure whose Switches are the IMPs, whose Pathways are the telephone lines that connect the IMPs, as well as the 1822 and VDH lines, and whose Hosts are the devices connected to the IMPs via either the 1822 or VDH interfaces. From the perspective of this Network Structure, the Pathways (telephone lines) have no internal structure, but rather are merely a passive and transparent medium. When we say that the Pathways have no internal structure, we mean simply that they contain no intermediate Switches (i.e., IMPs) and no Hosts of the particular Network Structure (the ARPANET) under discussion. Of course, this is quite a significant abstraction. What we regard as a simple wire (a telephone line) may in fact be no wire at all, but

an entire network, the telephone network! Within the telephone network, there may be any number of fancy computer switching devices, which might be just as complicated as the ARPANET's IMPs. From the perspective of the telephone company, one might want to regard each telephone line as a Network Structure, which contains exactly two Hosts (viz., the two IMPs at the end-points of the line). The Switches of this Network Structure are the telephone switching devices, and the Pathways really are just wires. Or, if we had reason to, we could regard the ARPANET as a sort of hybrid Network Structure, whose Switches included both the IMPs and the telephone company switching devices, and whose Pathways were wires terminating either at the IMPs or the other switching devices. As it happens, we generally don't find this last means of conceptual organization to be very useful. Since we have no control over, and little information about, the telephone company switching devices, it is most "convenient" not to have to think about them, but rather to just regard each telephone line as a simple Pathway, with no internal structure, and no intermediate Switches. It is important to understand that calling the ARPANET a Network Structure whose Switches are IMPs and whose Pathways are telephone lines does not beg any questions about how the telephone network really works; it is just a matter of imposing a conceptual organization that we find useful for some purpose.

Of course, the telephone lines are not the only Pathways in the ARPANET. Each (local or distant) host is also the endpoint of a Pathway, though one which really is a wire. In general, we will find it useful to distinguish those Pathways in a Network Structure which connect Host to Switch (ACCESS PATHWAYS) from those which connect Switch to Switch (INTERNAL PATHWAYS).

Another example of a Network Structure is one whose Switches are the gateways on the ARPA Catenet, whose Pathways are segments of ARPA-controlled networks, and whose Hosts are hosts on the networks which are part of the Catenet. Within this Network Structure, the gateways must be regarded as Switches, since they perform store-and-forward functions, and data from one host to another is routed through a sequence of gateways. Of course, a gateway, while a Switch within the Network Structure of the Catenet, may also be a Host within the Network Structure of the ARPANET. The proper classification of an entity is a matter of what function it performs within the concrete realization of some particular Network Structure; the same physical entity may perform different functions in different Network Structures of which it is a part.

We should be a bit more precise about the Pathways of the Catenet Network Structure. Suppose there are 4 gateways on the ARPANET. Then the gateways are fully connected through a set of 12 distinct Pathways (i.e., each gateway has a Pathway to each of

the other 3 gateways). Although each of the 12 Pathways utilizes the ARPANET, we really have 12 distinct Pathways, each of which has different characteristics as regards delay, throughput, and operability. That is why we said above that the Pathways of the Catenet should be identified with SEGMENTS of ARPA-controlled networks, rather than with the entire networks. Furthermore, each of the 4 Gateways has a distinct Pathway to EACH host on the ARPANET. That is, within the Network Structure of the Catenet, each of the hosts on the ARPANET (all of which are also Hosts on the internet) is MULTI-HOMED to each of the 4 Gateways via a distinct Pathway. IN PRINCIPLE, THIS MULTI-HOMING IS NO DIFFERENT THAN THE MULTI-HOMING OF A SINGLE (LOGICALLY ADDRESSED) HOST TO TWO DIFFERENT ARPANET IMPS (see IEN 183). As we shall see, regarding all the Hosts on the Catenet as being multi-homed to the Switches (i.e., gateways) of the Catenet is a very important feature of the network architecture we will propose for internetting. We will argue that many of the problems encountered in the current Catenet configuration are a result of the failure to properly conceive internet hosts as being multi-homed, and that the lessons learned from a study of how to do multi-homing on individual packet-switching networks can be applied fairly directly.

The "Network Structure" model is intended to be completely general. We can, for example, handle an arbitrarily nested hierarchical internet by establishing a Network Structure whose

Pathways are themselves complex internet configurations. We can also model overlapping internet configurations. Consider, for example, four machines, A, B, C, and D connected in order in a ring. In principle, we could treat this as two Network Structures, one in which A and C are Switches and B and D are Hosts, and another in which B and D are Switches and A and C are hosts. This might be useful if, for example, we have two different internets, with incompatible gateways, connecting the same set of packet-switching networks. The model should be general enough to accommodate complex configurations like this.

## 1.2 Deficiencies of the Old Model

The idea that the design of an architecture for the internet should be guided by the development of an abstract model is hardly original. The earliest IENs often considered the need for a model, but the discussion there seems to be at an insufficient level of abstraction. That is, there is much discussion of the need for a "model of a gateway," but no discussion of the need for a model of the internet as a gestalt, with a network architecture of which the gateways are only a part.

It is apparent though that gateway designers and implementers did work with an IMPLICIT model of the internet in mind. While the gateway was the focus of much discussion, however, little critical attention was focussed on the implicit model of the internet itself. This implicit model represents the



gateways as ordinary network nodes, and the component networks of the internet as ordinary network lines. Surprisingly, hosts are not represented at all in this model. (One may be tempted to think of a host as a sort of piece of one of the "lines" in this model, but remember that the lines are not supposed to have any internal structure.) The internet routing problem was conceived of as the problem of how to get data from one gateway through a sequence of intermediate gateways to a destination network (which, in the model, is a destination line!?).

Our experience with the Catenet should have made it quite clear by now that the basic idea underlying this implicit model is overly simplistic. This basic idea is that the end-user will know what network his destination host is attached to, and only needs the gateways to get his data somewhere or other (it doesn't matter where) on that destination network. At that point, the destination network is supposed to take over, and there is no further work for the gateways to do. We now know, of course, that things are not so simple. The way in which the gateways get traffic to the destination network may be very important. A particular host on a particular network might be reachable from one gateway on that network, but not from another gateway on that network. (This is the "partitioned net" problem.) Or performance considerations might dictate that a host be reached through one gateway in preference to another gateway on that same net. It should not be any surprise that this sort of problem has

proved very troublesome in the Catenet, for the problem is built right into the conceptual mechanism that guided the development of the Catenet. The fact that the old implicit model of the internet contains no representation at all of hosts makes it virtually impossible for gateways that were built with that model in mind to have any means of representing host-specific information. It also makes it virtually impossible to take advantage of (or even to take cognizance of) the way in which Hosts can be regarded as being "multi-homed" to the gateways. Failure to model the hosts also makes it difficult to handle the case of hosts which are not always connected to the same network (e.g., "mobile hosts"), or hosts which are connected to two or more networks.

Further difficulties arise from the way in which networks are represented as "lines." If a network is like a line, then it must be either up or down. There is no way to represent the fact that some "parts" of the "line" can be reached from only one end-point, but not the other. That is, it is difficult to make the internet system respond to peculiarities in the behavior or operational status of the underlying packet-switching networks, since much of that behavior has no analog in the world of telephone lines. Of course, it cannot really be claimed that problems like these can never be resolved at all within the current Catenet configuration, but only that possible resolutions will not fit easily into the Catenet's architecture, and so will

generally appear to be "kludges" or "hacks" which are grafted on in order to get around particular operational problems as they happen to arise. Perhaps a reading of some of the more recent IENs will bear out this impression.

In attacking the old implicit internet model, we are not simply trying to beat a dead horse, or to attack a straw man. Rather, we are just trying to emphasize that the way in which one initially models or thinks of a set of related problems will necessarily have a large effect on the way the problems are dealt with in the final system. A good model for the internet should provide a framework for discussion of internetting issues which enables us to place each issue in proper perspective, and which makes clear the inter-relationships among the various problems and proposed solutions to them. When important problems (such as the "partitioned net" problem) cannot even be stated in the terms of a particular model, it becomes clear that that model does not provide a proper framework for the discussion of the issues. A good model should also make it possible to evaluate the effect of various schemes as part of an integrated system, making it easier to determine whether or not some proposed scheme gives rise to more problems than it solves. By allowing us to see solutions to particular problems as part of an integrated system, the model also gives us a means of choosing among different schemes which seem to solve the same problem, since some of those schemes may fit more easily or more naturally into the integrated system than

do others. A good model should also suggest solutions to problems that have previously seemed very vexing (we shall argue, in subsequent papers of this series, for example, that representing hosts as being multi-homed to gateways suggests certain addressing schemes that might otherwise be overlooked, and also subsumes a number of problems previously thought unrelated under a common rubric). We believe that the model we proposed in the previous section offers a much more coherent framework; hopefully this will become apparent as we begin to work through the issues of internetting in this and subsequent papers.

### 1.3 The Importance of Pathway Characteristics

As we have already pointed out, the proposed new model of the internet as a Network Structure allows us to see the ARPANET itself as an internet, built upon pieces of the telephone network. In principle, then, the ARPANET is no different than the Catenet, which is an internet built upon pieces of the ARPANET, SATNET, and various other ARPA-controlled networks. Yet it does seem that the Catenet poses problems which are more difficult and less tractable than are the problems posed by the ARPANET. Why should this be the case, if the ARPANET and the Catenet are both internets of a sort? The answer seems to lie in the characteristics of the individual networks that constitute the Pathways in these two different "internets." The pieces of

the telephone network which transmit data within the ARPANET do so with well-known and well-understood (indeed, with constant) delay characteristics. The capacity of these pieces of the telephone network is also a constant. Many of the ARPANET's protocols and algorithms (both internally and at the host access level) make use of these facts, and break down when applied to Pathways of significantly different characteristics. Even within the ARPANET, we have seen the importance of modifying our protocols and algorithms to take account of differing Pathway characteristics. For example, the line up/down protocol which each pair of neighboring IMPs runs together to determine the operational status of the line connecting them must be tuned differently for lines of differing bandwidths or propagation delays. Particular difficulty has been encountered in making this protocol work properly over "line 77," the "transparent" connection via SATNET to London. One problem in trying to extend ARPANET-like protocols and algorithms to the Catenet environment is to come to a better understanding of how those protocols and algorithms actually depend on assumptions about the Pathway characteristics. Since many of these assumptions may be implicit, and nowhere clearly stated, this is not a simple problem. As we develop our proposed internet architecture, we will try to emphasize the role played by assumptions about Pathway characteristics.

(Although we will be primarily concerned with protocols and algorithms to be used in the actual day-to-day operation of the internet, it is worth noting that the variability of the Pathway characteristics of the internet also has implications with respect to the topological layout of the internet. When designing the topology of a packet-switching network, one often makes use of mathematical tools (often automated) for optimizing the topology with respect to some cost-function, such as delay. These mathematical tools are based on particular mathematical models of networks which in turn are based on results from queuing theory which assert a particular relationship between delay and load over telephone lines. Whatever the merits of those mathematical models (and there is much to question in them), they will not be applicable at all to the topological design of the internet. When a Pathway is a packet-switching network, rather than a telephone line, it is probably meaningless even to ask what its bandwidth is, since it just does not have a constant bandwidth. That is, the maximum throughput that can be obtained between two gateways over a particular packet-switching network will vary quite a bit over time, and will depend on what happens to be going on within that network. In addition, the relationship between delay over a packet-switching network and the offered load is much more difficult to model mathematically than is the same relationship over a telephone line. Issues of how to properly lay out the topology of the internet to obtain

best performance or least "cost" probably will not be able to be dealt with in any systematic way until we have much more experience with day-to-day internet operations.)

The gateways which are currently deployed on the Catenet do not seem to take Pathway characteristics into account at all. Certainly there has been no attempt to optimize the gateways to the particular packet-switching networks that they are connected to, or even to make the gateways take proper notice of the various protocol messages that the networks will send to the gateways. To some extent, gateways really do seem to treat packet-switching networks as mere wires, simply throwing the bits at the network interface, and not dealing with the various exception states that continually arise when attempting to access a network. One of the main themes that we shall be developing is that A ROBUST AND HIGH-PERFORMANCE NETWORK STRUCTURE JUST IS NOT POSSIBLE UNLESS THE SWITCHES ARE CAREFULLY TUNED TO MAKE THE MOST EFFICIENT POSSIBLE USE OF THE PATHWAYS. As we have stated above, the ARPANET IMPs often have to be tuned to the particular characteristics of different telephone lines, and it is that much more important for the gateways to be tuned to the particular characteristics of the packet-switching networks that serve as Pathways between them. It is important to understand that this sort of issue does not apply only to the way in which gateways use the INTERNAL PATHWAYS of the internet, but also to the way in which hosts and gateways use the ACCESS PATHWAYS of the internet.

We will develop these issues in much more detail in subsequent papers.

We have claimed that the only reason we tend to regard the telephone network as a Pathway with no internal structure is that we find it "convenient" to do so. If we are going to properly apply the Network Structure model to the ARPANET, the Catenet, or any other networking or internetting environment, it is important to come to a clear understanding of just when it is and is not "convenient" or "useful" to ignore the internal structure of a communications medium, and model it as a Pathway. (Though ignoring the internal structure of a communications medium is NOT the same thing as ignoring its delay/throughput/reliability characteristics; as we shall repeatedly emphasize, there are no conditions under which it is possible to do that without incurring extreme penalties in efficiency and/or reliability.) There are basically two good reasons why we might want to ignore the internal structure of a communications medium:

- 1) Efficiency. Some algorithms or protocols in the Network Structure may need to be driven off some sort of model or representation of the network. For example, the SPF routing algorithm in the ARPANET has a database which represents the network topology. (We will often use the term "SPF routing" to refer to the ARPANET's current routing algorithm [1,2], in contradistinction to the



ARPANET's original routing algorithm [3]. The Catenet's current routing algorithm is based on the original ARPANET routing algorithm, but internetters should be aware that that algorithm had a number of serious deficiencies, especially under heavy load, and was removed from the ARPANET over two years ago, to be replaced with SPF routing.) Dividing a single network into several different Network Structures, and representing some of those as Pathways with no internal structure, may be necessary if we need to keep a bound on the size of the database while allowing the actual physical configuration of the network to grow without bound. This is one of several important considerations driving the internet development.

- 2) Partial transparency. One may want to be able to replace the networks which underlie certain Pathways without having to make extensive changes to the software of the Switches. Or one may simply not be able or willing to get any information about some underlying network. Either of these is a good reason to try to treat that underlying network transparently. Note, however, that the best that we can hope to achieve is PARTIAL transparency. If a Pathway is replaced by another Pathway of very different characteristics, we cannot realistically expect to maintain efficient performance

without modifying the Switches in some way to take account of the new characteristics. However, it may be possible to restrict the magnitude of the changes; for example, perhaps only the software in the Switches which are the endpoints of the Pathways will need to be changed. This is an issue that we will have to consider in great detail; certainly it is one of the most important considerations driving the internet effort.

Note that these considerations, important as they may be, are really just matters of degree, and generally must be traded off against other considerations. We can ignore the internal structure of a Pathway to a greater or lesser degree. It is possible, for example, that we will want our internet gateways to exchange information with certain ARPANET IMPs, even though in general we want the internet gateways to be able to ignore the internal structure of the ARPANET. The principle of ignoring the internal structure of a Pathway is supposed to be a tool to help us, not a straitjacket to prevent us from getting needed information. This is another issue to which we shall return repeatedly in subsequent papers of this series.

One of the aspects of a Network Structure that is most sensitive to Pathway characteristics, and to the decision to ignore the internal structure of a Pathway, is its MAINTAINABILITY. Maintainability is one of the most important,

though most neglected, areas of network design. In the long run, the ability to maintain the network (which means the ability to isolate faults and to repair them) can have a much larger effect on the network's efficacy as a communications utility than almost any other consideration. In some sense, maintainability is the "bottom line;" if a network is subject to repeated inexplicable failures and degradations, then the users will be driven away, and all the effort that has gone into careful optimizations of the algorithms and protocols will have been wasted. In a complex Network Structure, which may include Pathways of arbitrary internal complexity, maintainability is a very crucial issue. A good example of the kinds of problems that can arise may be taken from our experience with the ARPANET's line 77 (the "transparent" connection to the London-TIP via SATNET). The ARPANET generally treats this as if it were a telephone circuit, even though it actually consists of a large number of terrestrial access lines, SIMPs (SATNET nodes), and satellite broadcast facilities, any component of which might fail in its own peculiar way. When this special connection was installed, the intention was that it be as transparent as possible to the ARPANET. It turns out, however, that a side-effect of treating SATNET transparently is that IT ALSO BECOMES TRANSPARENT TO THE FAULT-ISOLATION TECHNIQUES WHICH ARE USUALLY USED FOR TROUBLE-SHOOTING ARPANET LINES. That is, the usual fault isolation techniques do not see into the structure of this "line", and hence can say no more than that the

line is up or down. While this is a consequence of the transparent design, it causes great difficulty, especially since the various components of this line are maintained by different organizations, each of which prefers to believe that the problem is someone else's responsibility. Furthermore, since the IMPs at each end of the line (which are also Hosts on the Network Structure of SATNET) don't realize that they are actually accessing another network, and don't use the usual network access protocol for SATNET, some of SATNET's standard fault isolation techniques are bypassed too. A problem that has been recurring with some frequency is that the ARPANET's line up/down protocol just will not bring the SATNET "line" up, even though SATNET itself seems to be working fine, according to the independent SATNET monitoring. At one time, the team of ARPANET and SATNET people working on the problem originally seemed to be in agreement that the source of the problem was in the design of the ARPANET's line up/down protocol. On further investigation, however, it turned out that the real problem was that the terrestrial access line between the London-TIP and one of the SIMPs really was experiencing intermittent failures, and hence that the ARPANET's protocol had been performing correctly when it refused to bring the SATNET "line" up. However, since neither the SIMP nor the London-TIP (really a host on SATNET) treated this access line as SATNET-host access lines are normally treated, the SATNET people found it very difficult to test this

line by itself in order to do fault isolation. If we have a Network Structure whose Pathways can themselves be arbitrarily complex and nested internets, then this sort of problem can be expected to arise with great frequency, UNLESS PROPER INSTRUMENTATION AND FAULT ISOLATION MECHANISMS ARE DESIGNED INTO THE NETWORK STRUCTURE FROM THE VERY BEGINNING. To some extent, some of these problems may just be inevitable once we decide to ignore the internal structure of a Pathway. The extent to which this is or is not true is a very important issue for us to consider, since it may ultimately be one of the most important factors in determining whether a reliable, operational, flexible, and growing internet configuration is really feasible. We will try to keep maintainability issues in mind throughout our entire discussion of the internet architecture that we propose.

To a lesser extent, this sort of problem can occur even purely within the ARPANET. Since ARPANET links are really pieces of the telephone network, it is worth asking whether the transparency of the telephone network impacts our ability to maintain the ARPANET. In fact, this does sometimes cause problems. When the ARPANET Network Control Center complains to the telephone company that a line is not operational, they really cannot provide the telephone company with very much data as to what is really wrong. Sometimes it takes the telephone company a long time to fix the problem, and it is not uncommon for the telephone company to claim that a line is fixed and to return the

line to the ARPANET, after which it is discovered that the ARPANET's line up/down protocol still will not bring it up (because the packet error rate is too great). Yet the situation is generally acceptable -- the ARPANET itself does a good enough job of detecting when there are problems with the lines, and the phone company does a good enough job (generally) of fault isolation within their own network to be able to fix problems relatively quickly. However, in a Network Structure whose Pathways consist of packet-switching networks, rather than telephone lines, we probably wouldn't be so lucky. The more complex and poorly understood the Pathways actually are (and the behavior of packet-switching networks, not to mention internets, is still quite poorly understood), the less likely it is that a simple complaint to the maintainer of the Pathway will get timely results. As the Pathway characteristics become more and more complex, it will become more and more important to have instrumentation at all levels, including the Switches and Hosts at Pathway end-points, to aid in diagnosing possible problems. As much as we might want to be able to regard the Pathways as fully transparent, it may turn out that the sort of instrumentation needed to help in fault isolation is dependent on the particular characteristics of the Pathway. This is another issue that we will have to keep in mind throughout.

Wherever possible, as we develop our proposed internet architecture, we will try to "parameterize" the effect of Pathway

characteristics on the robustness and performance of the architecture. It will certainly be important to know whether it is really possible to build a robust high-performance Network Structure out of Pathways with totally arbitrary characteristics (probably not), and if not, just how many constraints we must put on the Pathway characteristics in order to get reasonable performance out of our architecture. This approach will help us to get some understanding in advance of how large and varied a configuration our architecture can support. This approach will also help us to understand how our architecture can be adapted to other applications in which we can place more constraints than would be appropriate for the Catenet. Consider, for example, a Network Structure all of whose Pathways are versions of the ARPANET which are largely homogeneous and under the control of a single organization. Within this Network Structure, we might be able to introduce much more effective and efficient routing and congestion control mechanisms than in a Network Structure whose Pathways consist of many different kinds of networks controlled by many different organizations with varied interests (e.g., the Catenet). In fact, this rather homogeneous Network Structure might not properly be called an "internet" at all; it might just be regarded as the ARPANET with area routing. ("Area routing" refers to any routing scheme in which a network is divided into several areas, such that Switches in each area have explicit routes only to other Switches in the same area, but not to

Switches in other areas. Routes are available for getting to other areas, but the internal structure of these other areas is disregarded. The term is usually applied to routing schemes used in single homogeneous, packet-switching networks, as opposed to internets, however.) One of the advantages of our model is that it enables us to see internetting and area routing as applications that differ only in regard to the Pathway characteristics of the appropriate Network Structure.

#### 1.4 A Functional View of the Internet

Let's look now at how we might model the OPERATION of a Network Structure. There seem to be four main steps involved in getting data from a source Host to a destination Host:

- 1) A source Host submits a message to a source Switch, providing it with the name of a destination Host to which the message should be delivered, as well as with any other information which is needed to specify necessary constraints on the characteristics of data delivery.

(Note that we said "the NAME of a destination Host", not the address of a destination Host. We will return to this very important point in later papers.)

- 2) The source Switch must map the name of the destination Host into the address OF A DESTINATION SWITCH. This may or may not be identical to the source Switch itself. If



the name of the destination Host maps to several possible destination Switch addresses (multi-homing), the source Switch must choose one, and this must be one which has a currently operational Pathway to the destination Host.

3) Using the routing scheme of the Network Structure, the source Switch must get the message to the destination Switch, via some (possibly empty) sequence of intermediate Switches.

4) The destination Switch must get the message to the destination Host.

This model of operation attempts to make a clear separation between the protocols which the source and destination Hosts must use to access the Network Structure (steps 1 and 4), the protocols which the Network Structure uses internally to move data from one point to another (steps 2 and 3), and the protocols used by the Hosts to talk to each other. This separation is required by the precepts of protocol layering, and also by common sense, since only with a clear separation of access functions from internal functions can we maintain the flexibility to make internal changes in the Network Structure without impacting the access software in the Hosts. The need for this separation between access functions and internal functions is taken for granted in most ordinary packet-switching applications, but has not been incorporated at all into the current Catenet. The

current Catenet protocols freely intermix access functions with internal functions, and in fact there is only a single protocol, IP, which contains elements needed for Hosts to talk to each other, for Hosts to talk to Switches, and for Switches to talk to Switches. This seems to be a consequence of the old idea that the internet is just a series of networks pasted together by hosts which are on two or more of the networks. That idea makes it difficult to distinguish the gateways from the hosts, or to properly take account of the fact that although gateways are Hosts in some Network Structure (that of the individual packet-switching networks), they are also Switches in another (that of the internet). A proper distinction of access functions from internal functions seems essential, though, if we are to build an internet which functions as a real network, rather than as a series of pasted together networks and gateways.

Any Network Structure which is to be operational must have some way of performing these four steps. Furthermore, in order for particular applications to get satisfactory performance out of their use of the Network Structure, the Network Structure must perform these functions under certain constraints with respect to delay, throughput, reliability, sequential delivery, and fault detection. (By "fault detection", we mean the ability of the Network Structure to inform the user about exceptional conditions, such as the destination host's being down or unreachable. This constraint is often neglected in the design of

network architectures or protocols, but seems quite important in a robust operational environment.) The ability to gather and report exception information at various levels of protocol is important both for system maintenance and for general user satisfaction. Nothing makes a worse impression on a user than a mysterious degradation in service about which he can get no feedback. It is not immediately obvious, though, to what extent the various protocols used to operate a Network Structure must ensure that these constraints are met. It is generally understood that if some application requiring inter-process communication must place constraints (such as sequentiality) on the characteristics of the communications, then the application itself must utilize some high level protocol (at the Host-Host level, or even higher) which will enforce the constraints, rather than depending on the low-level communications medium to enforce them. What seems to be less generally understood is the fact that, in general, and other things being equal, the performance which some user gets from his high-level protocol will be better if the lower level protocols pass data to the high level protocols in a way which already satisfies the constraints that the high level protocols must impose. For example, an application which requires sequentiality will have to use a high level protocol like TCP which guarantees sequentiality. However, the end user will generally see much better performance, in terms of throughput and delay, if the protocols below TCP maintain

sequentiality, since this will require TCP to do less work, and put less of a drain on the resources (such as buffer space, sequence number space, and host CPU bandwidth) needed by TCP. The area of fault detection and reporting provides a good example here. A user attempting to talk to a dead host might find his TCP typing the message "excessive retransmissions" to him. This sort of message would not generally be too useful to a user since, if he is not a network expert, it gives him no clue of what the actual situation is. The average user might prefer to know that the destination host is dead, so he can try again later, but this information is very difficult, if not impossible, to obtain solely at the TCP level, although it might be quite simple to obtain at a lower protocol level. Of course, putting more functionality in lower level protocols also has a cost, as well as a potential benefit, and costs often trade off with benefits in surprising ways. As we shall see, producing an operational Network Structure requires a large number of protocols, and we will attempt to deal with considerations like these as we consider specific protocol issues. Not surprisingly, we will find that cost-benefit trade-offs for both access protocols and internal protocols will often depend on the characteristics of the Pathways in the Network Structure.

## REFERENCES

1. J.M. McQuillan, I. Richer, E.C. Rosen, "The New Routing Algorithm for the ARPANET", IEEE TRANSACTIONS ON COMMUNICATIONS, May 1980.
2. E.C. Rosen, "The Updating Protocol of ARPANET's New Routing Algorithm," COMPUTER NETWORKS, February 1980.
3. J.M. McQuillan, G. Falk, I. Richer, "A Review of the Development and Performance of the ARPANET Routing Algorithm," IEEE TRANSACTIONS ON COMMUNICATIONS, December 1978.