Postel

INDRA
Working
Paper

INDRA Note 743
IEN 83
23rd March 1979

Addressing Through Port Expanders

P. T. Kirstein

C. J. Bennett

ABSTRACT: This note examines the difficulties which have been encountered in addressing hosts and gateways through port expanders. It attempts to identify the fundamental characteristics of the problem and principles of solution. Two cases of particular interest – ARPANET IMPs and Internet Gateways – are examined in detail.

Dept of Statistics & Computer Science
University College, London

Addressing Through Port Expanders

P. T. Kirstein

C. J. Bennett

ABSTRACT: This note examines the difficulties which have been encountered in addressing hosts and gateways through port expanders. It attempts to identify the fundamental characteristics of the problem and principles of solution. Two cases of particular interest - ARPANET IMPs and Internet Gateways - are examined in detail.

Dept of Statistics & Computer Science
University College, London

1. INTRODUCTION

Since the port expander was introduced into the internetwork project, continual problems have arisen in connection with addressing devices which lie at the end of its lines. Two different solutions have been proposed at various times. In the early port expanders, each port was addressed using internet headers in which it was identified by being given a separate network number. This was always recognised to be an unsatisfactory hack, and at the Internet Meeting held in November 1978, a new solution was adopted for ARPANET whereby the mapping of ARPANET host IDs to Internet host IDs included a 'logical host' field, which a port expander could use to select a port on the basis of some locally convenient algorithm. This solution is clearly better than the first, but it still suffers from two major flaws.

1) It cannot be used to address transit gateways if there is more than one gateway on the port expander (or if there are any gateways on other port expanders connected to the first in a daisy chain fashion). This is because the internet header in a packet contains the address of the ultimate destination of the packet, and not the internet address of the gateway. If there is only one gateway, it can be handled by defaulting any unrecognised internet address to that gateway's port.

2) Both solutions require that all but one of the hosts attached to the port expander (all, if one of the hosts is a gateway) be able to process internet headers simply so that the addressing problem can be handled. In all other respects, a host talking to an ARPANET IMP through a port expander can act as an ordinary ARPANET host. It may not be particularly necessary that the host be an internet host. For example, UCL's SYSTIME 5000 could come up on the ARPANET tomorrow as an NCP host, which would be very desirable for many reasons, but this cannot be done because it would be sitting on a port expander and hence has to have an internet address. No doubt many ARPANET sites would like to increase the number of hosts supported by an IMP, but the overhead of implementing internet software prevents them from using port expanders to do so.

In other words, neither proposal gives a completely satisfactory solution. It will be argued below that this is because they are based on using internet headers to remedy an addressing deficiency that is not necessarily connected with internet problems. The fact that the solutions are inadequate, and that correct solutions have not been found, suggests that the problem has not been correctly defined or understood by the Internet community. The aim of this note is to identify the fundamental
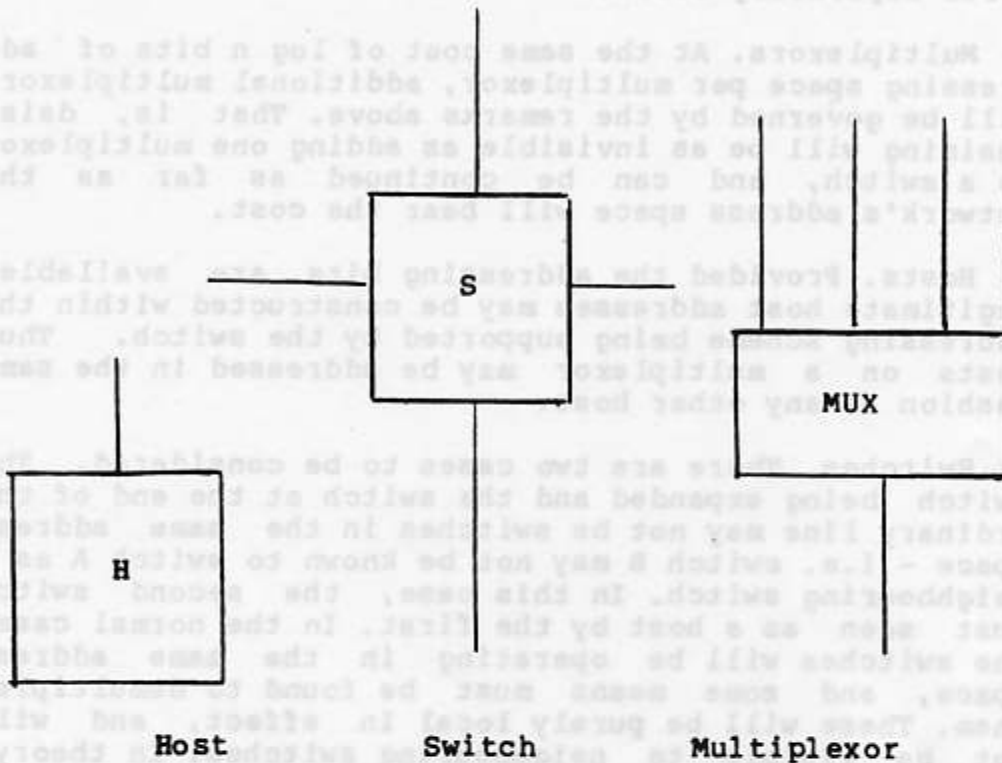
characteristics of the problem, and to suggest solutions that will behave correctly in forseeable configurations.

2. FUNCTIONAL ELEMENTS

In the following discussion, we will be concerned with three functionally different entities. In order to avoid getting entangled in arguments related to specific systems rather than generic properties, these entities will initially be given names which hopefully do not have any of the connotations of the names used for them in specialised contexts (in particular, the Internet context). The properties are illustrated in figure 1.

The three entities involved are:

HOSTS. A host is an entity which acts as a traffic source and/or sink. A message which is transmitted by a host was created by some process within that host. A message received by a host will be processed in its entirety within the host.

SWITCHES. A switch is an entity which receives messages on any of several input lines, and transmits the same message (with the possible exception of some processing of the header) on any of several output lines. Two examples of switches are ARPANET IMPs and CATENET Gateways.

MULTIPLEXORS. A multiplexor is a very specialised kind of switch, which distinguishes one line from the n others it has. All incoming traffic on the special line is always sent to one of the ordinary lines. All incoming traffic on the ordinary lines is sent to the special line. An example of a multiplexor is the port expander.

Figure 1: Hosts, Switches and Multiplexors

A multiplexor can act in association with a switch to give the effect of increasing the number of lines available to that switch. In this case, the 'special' line is the line connecting the multiplexor to the switch it is expanding. The multiplexor in such a case should be invisible both to the switch, which will see either one or n entities on that physical port, and to the entities on the n output lines, which will simply see the switch. Thus the port expansion effect is purely local, and should not normally affect the end-to-end communication between hosts.

The cost of this is to add n entities to the address space being supported by the switch being expanded. That is, log n bits (to the base two) must be added to the address field, or the equivalent of that many free bits be found within it. If free combinations are to be found, their distribution will depend on the construction rules for addresses within the address space.

Any of the three entities above may be found at the other end of the multiplexor's ordinary lines. The way the switch will perceive these will differ depending on the entity. These are considered separately below.

1) Multiplexors. At the same cost of log n bits of addressing space per multiplexor, additional multiplexors will be governed by the remarks above. That is, daisy chaining will be as invisible as adding one multiplexor to a switch, and can be continued as far as the network's address space will bear the cost.

2) Hosts. Provided the addressing bits are available, legitimate host addresses may be constructed within the addressing scheme being supported by the switch. Thus hosts on a multiplexor may be addressed in the same fashion as any other host.

3) Switches. There are two cases to be considered. The switch being expanded and the switch at the end of the ordinary line may not be switches in the same address space - i.e. switch B may not be known to switch A as a neighbouring switch. In this case, the second switch must seen as a host by the first. In the normal case, the switches will be operating in the same address space, and some means must be found to demultiplex them. These will be purely local in effect, and will not be visible to neighbouring switches. In theory, they could even vary from switch to switch. The cost is again log n bits of addressing space, where n is the number of switches handled through the port expander.

These various situations are illustrated in figure 2.

The principle point to note is that an entity on a port expander exists within the address space of the switch being expanded, at a cost of log n bits. Thus, an ARPANET IMP expander SHOULD be supporting ARPANET hosts (c.f. point 2 in the introduction to this note); a gateway port expander is supporting INTERNET hosts; a SATNET port expander SATNET hosts, and so on. The remainder of this note will consider the ARPANET and INTERNET cases in more detail, as the application of the principle is not totally straightforward in these cases.
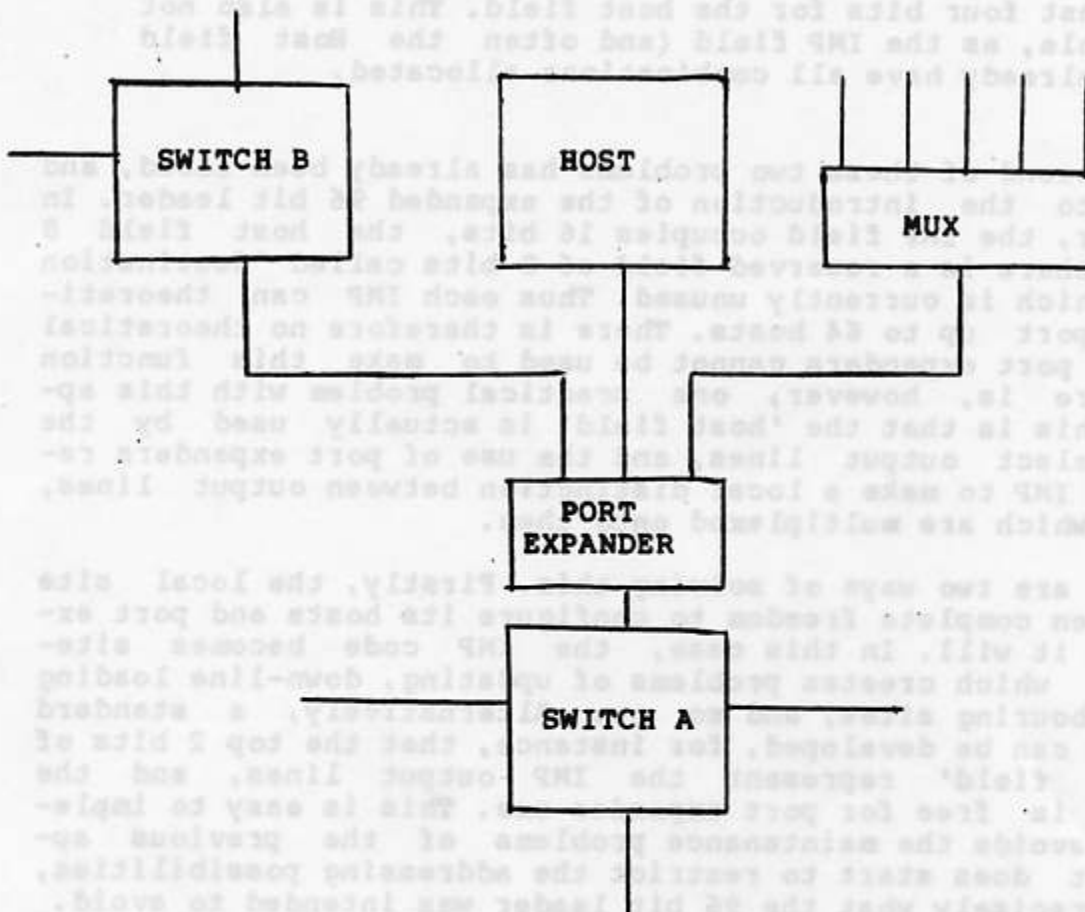
```
┌──────────────┐        ┌──────────────┐      │  │  │  │
│   SWITCH B    │        │     HOST     │   ┌──┴──┴──┴──┴──┐
└──────────────┘        └──────────────┘   │     MUX      │
                                            └──────────────┘

                        ┌──────────────┐
                        │     PORT     │
                        │   EXPANDER   │
                        └──────────────┘

                        ┌──────────────┐
                        │   SWITCH A    │
                        └──────────────┘
```

Figure 2: Possible Port Expansion Relationships.

3. ARPANET IMP EXPANSION

The addressing difficulties faced by port expansion within the ARPANET are caused by two features of the normal, 32 bit leader, ARPANET addressing scheme.

1) An ARPANET address is very highly stuctured. It closely reflects the physical configuration of the network, and this fact is embodied in much low-level software. Subfields exist for IMPs and hosts. In the 32 bit leader, these are 6 bits wide and 2 bits wide respectively. A port expander supporting up to 8 entities (always seen as hosts) will require a three bit host field at least.

2) Two approaches could be adopted. The 32 bit leader

could be augmented, which is not possible as the bits are not available; or it could be restructured to allow at least four bits for the host field. This is also not possible, as the IMP field (and often the Host field too) already have all combinations allocated.

The second of these two problems has already been faced, and has led to the introduction of the expanded 96 bit leader. In this leader, the IMP field occupies 16 bits, the host field 8 bits, and there is a reserved field of 8 bits called 'destination network' which is currently unused. Thus each IMP can theoretically support up to 64 hosts. There is therefore no theoretical reason why port expanders cannot be used to make this function work. There is, however, one practical problem with this approach. This is that the 'host field' is actually used by the IMP to select output lines, and the use of port expanders requires the IMP to make a local distinction between output lines, and hosts which are multiplexed onto them.

There are two ways of solving this. Firstly, the local site can be given complete freedom to configure its hosts and port expanders as it will. In this case, the IMP code becomes site-dependent, which creates problems of updating, down-line loading from neighbouring sites, and so on. Alternatively, a standard convention can be developed, for instance, that the top 2 bits of the 'host field' represent the IMP output lines, and the remainder is free for port expander use. This is easy to implement, and avoids the maintenance problems of the previous approach. It does start to restrict the addressing possibilities, which is precisely what the 96 bit leader was intended to avoid.

The alternative solution, proposed here, is to allocate the 'destination network' field to port expansion. This field is currently unused, and given the direction of the internet project it is unlikely that it will ever be used for internetwork purposes. In this solution, the IMP will route the packet to an output line based on the destination host field; if the entity here is in fact a port expander, further demultiplexing is done based on the 'network' field. This approach has a number of advantages:

1) Few changes, if any, are required to current IMP software. The IMP does not currently examine this field, and there is no reason why it should do so. The only possible exception to this is in connection with host status messages. Possibly, even these could be handled by the port expander, with the exceptional case of port expander collapse being handled normally by the IMP.

2) Port expander configurations can be done in any site-dependent fashion, and be completely invisible to both the local IMP and the ARPANET as a whole.

3) Except for the changes needed to handle 96 bit leaders (which may be required anyway), and the addressing changes, hosts on port expanders can use standard ARPANET software - they are, in effect, normal ARPANET hosts.

The disadvantages are:

1) Hosts not supporting 96-bit leaders will not be able to communicate with these hosts. Since current port expanders are only planned to support 96 bit leaders, this is nothing new.

2) ARPANET port expanders will need considerable recoding to remove the current internet dependencies. This is a temporary delay only.

Note that up to this point we have scarcely mentioned the internet program. This is deliberate. Under the scheme proposed here, the hosts on a port expander can be an NCP host, a gateway, or the planet Mars; the only requirement is that it support the 1822 interface. Since port expansion is a local, IMP based function, this is as it should be. However, it is worth saying a few words on how this scheme affects the internet program.

As we saw in the introduction, gateways connected to the ARPANET through port expanders currently exist in a kind of addressing limbo, from which they can only be rescued if they are the only gateway on the port expander. Under either of the schemes proposed in this section, a gateway is a proper ARPANET host with a well-understood ARPANET address; hence intergateway communication is achieved in the correct manner, by using local network protocols (in this case the ARPANET). Of course the intergateway messages must conform to internet standards but this is invisible from inside the ARPANET.

The mapping between internet addresses and ARPANET addresses may be subject to change. If the use of 96 bit leader's 'network' field is adopted, this corresponds very well to the current 'logical host' field - a one-to-one mapping betwen them is straightforward. The only foreseeable problem is that the 'IMP' field in the internet header is restricted to 8 bits, whereas the 96 bit leader allows 16 bits. IMP number 256 is going to cause problems with internet addressing in this scheme. However, despite past experience, we feel it is reasonable to assume that ARPANET will not in fact grow this large. If it does, then a more sophisticated ARPANET-Internet address mapping must be devised.

If hosts on port expanders are addressed using the ordinary host field, then the 'logical host' concept can be removed from the internet address mapping, to be replaced by a division of 8 bits host field and 16 bits of IMP field. In this way, the Internet address scheme will not suffer the embarassing position just described.

Finally, a comment on the current proposals for combining port expanders and gateways. While the next section will show that a gateway port expander is a perfectly legitimate object, it should be clear from the previous discussion that we feel combining an IMP port expander and a gateway is a misguided project. It is, of course, perfectly possible - the gateway would sit on a virtual port - but there is no common function between them, and hence no great advantage in doing it. Indeed, causing the same hardware to perform two unrelated communication services may well cause undesirable interactions between them, especially when one is being heavily used.

4. GATEWAY PORT EXPANDERS

In principle, gateway port expansion operates in a similar fashion to IMP port expansion. The major difference is that the 'lines' made available by the port expander are virtual gateway to gateway links (or gateway to internet host links). For a local switch inside the network the port expander is 'actually' the gateway host, and the internet port expansion function is totally invisible to it. Figure 3 illustrates this situation, showing both the physical links, and the logical port expansion involved for one hypothetical case.
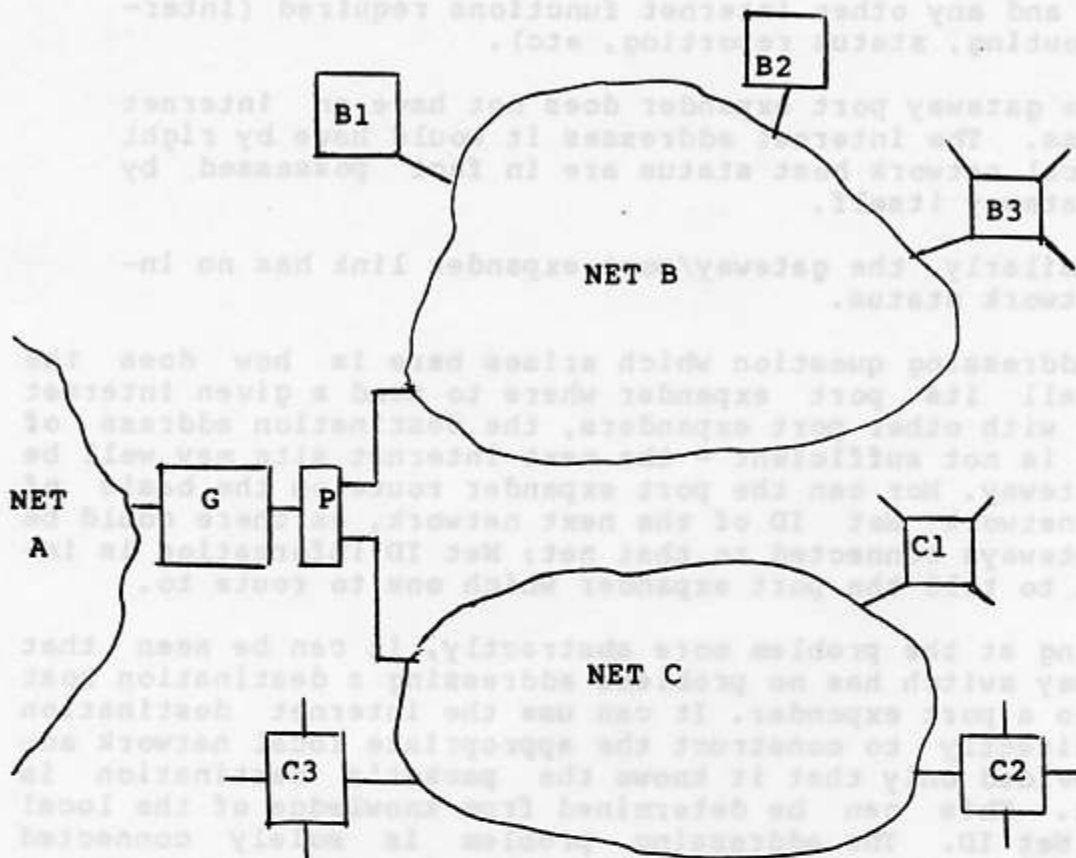
Figure 3: Gateway Port Expansion

There are a number of points to note about this configuration.

    1) The gateway itself is not seen as a host on any of the networks to which it is connected through the port expander. Accordingly the port expander must support the various local network protocols for these nets. Each line driver is supervised by software which regards itself as a local host frontending the gateway machine, and unaware it is coresident with others.

    2) The gateway/port expander link is therefore not constrained by any local network. Accordingly, this link protocol can be chosen for convenience.

    3) The gateway performs the internet switching func-

tion, and any other internet functions required (inter-
net routing, status reporting, etc).

4) The gateway port expander does not have an internet
address. The internet addresses it would have by right
of local network host status are in fact possessed by
the gateway itself.

5) Similarly, the gateway/port expander link has no in-
ternetwork status.

The addressing question which arises here is how does the
gateway tell its port expander where to send a given internet
packet. As with other port expanders, the destination address of
the packet is not sufficient - the next internet site may well be
another gateway. Nor can the port expander route on the basis of
the internetwork Net ID of the next network, as there could be
several gateways connected to that net; Net ID information is in-
sufficient to tell the port expander which one to route to.

Looking at the problem more abstractly, it can be seen that
the gateway switch has no problems addressing a destination host
attached to a port expander. It can use the internet destination
address directly to construct the appropriate local network ad-
dress, provided only that it knows the packet's destination is
that host. This can be determined from knowledge of the local
network's Net ID. The addressing problem is solely connected
with addressing the next internet switch - i.e. the next gateway.
An equivalent problem would be an ARPANET IMP trying to route
data to one of several neighbouring IMPs connected through the
port expander (as we saw, this problem does not in fact arise in
the ARPANET).

Thus, we can break the problem down into two parts: How do
internet switches normally address each other when routing inter-
net packets; and what differences are made by the presence of a
gateway port expander. The answer to the first part is that gate-
ways use the local network host ID's. However, the gateway and
the port expander communicate using some private protocol, and
the gateway is not strictly part of any network. Thus the gateway
cannot use any particular network's addressing conventions to in-
struct the port expander which output line to use.

The solution to this problem lies in the private
gateway/port expander protocol. A header field must be included
which the port expander can use to decide which output network to
send the message to, and to which host on that network. The size
and structue of this field is a local problem, as it is invisible
to the outside world. An alternative way of describing this is to
say that the gateway/port expander configuration form a hidden
transit network.

The following example (based on figure 3) shows three ways this scheme could work in one configuration.

Consider a gateway G connected to three networks, A, B and C. The connection to network A is direct, but to networks B and C the connection is through a gateway port expander P. On network B there are three other hosts with internet capability, B1, B2 and B3. B1 and B2 are destination hosts, B3 is another gateway. On network C, hosts C1, C2 and C3 are all gateways.

Messages to be sent through G to net A are handled in a straighforward fashion. They are sent through nets B or C to P, which in its guise as a local host performs any local network functions (acknowledgements etc) which are required. P then routes the packet to G which performs any internetwork functions required before routing the packet to net A. No addressing function is required for traffic being sent from P to G.

Messages being sent from net A through nets B or C, or from net B to net C, or vice versa, will require G to provide P with local information to decide where to send the packet next. Three approaches could be adopted:

1) A field of at least 3 bits is inserted before the internet header. These three bits are used to distinguish hosts B1, B2, B3, C1, C2 and C3 without regard to network ID or internet function performed by these hosts. The port expander chooses an output line driver based on some table look up, and passes the local network ID to the appropriate local network software.

2) The same field of at least three bits is structured to distinguish networks B and C. For example, bit 0 distinguishes B and C, and bits 1 and 2 distinguish the hosts within B and C. The port expander then chooses an output network line driver based on bit 0; the driver finds the local host based on bits 1 and 2.

3) The same field of at least three bits is structured to distinguish between internet hosts and internet gateways. A reserved value (eg 0) is interpreted by the port expander to mean "get the net and host ID from the internet ID field". Other values for the field give local gateway IDs according to schemes 1 or 2. In this example, both hosts B1 and B2 would be covered by the value 0; B3 could take the value 1, and C1 through C3 could take the values 4 to 6, using scheme 2.

Doubtless other schemes could be developed. The main point of the illustration is that hosts B1-3 and C1-3 could not tell the difference between them, and in fact would not be aware that

this mapping was being performed.

5. CONCLUSIONS

The main conclusion of this note is that port expansion is a function which is performed locally to the switch being expanded and should be visible only to the switch and the port expander itself. In addressing terms, log n bits of addressing space supported by the switch is required to handle the extra destination hosts provided. Switches are handled by a private switch/port expander protocol.

The two examples considered illustrate these points. ARPANET port expansion requires extra addressing bits which cannot be found in the 32 bit leader, but can in the 96 bit leader. Both solutions discussed here avoid the problems which arise with existing port expansion schemes. Gateway port expansion requires a private addressing protocol between the port expander and the gateway it is expanding so that other gateways can be addressed, but no problem exists with internet hosts.