Gateways, Architectures, and Heffalumps

M.A. PADLIPSKY
THE MITRE CORPORATION
Bedford, Massachusetts

ABSTRACT

The growth of autonomous intercomputer networks has led to a desire on the part of their respective proprietors to "gateway" from one to the other. Unfortunately, however, the implications and shortcomings of gateways which must translate or map between differing protocol suites are not widely understood. Some protocol sets have such severe functionality mismatches that proper T/MG's cannot be generated for them; all attempts to mesh heterogeneous suites are subject to numerous problems, including the introduction of "singularity points" on logical connections which would otherwise be able to enjoy the advantages of communications subnetwork alternate routing, loss of functionality, difficulty of Flow Control resolution, higher cost than non-translating/mapping Gateways, and the necessity of re-creating T/MG's when a given suite changes. The preferability of a protocol-compatible internet is also touched upon, as is the psychology of those soi-disant architects who posit T/MG's.

Gateways, Architectures, and Heffalumps

M. A. Padlipsky

In our collective zeal to remain (or become) abreast of the State of the Art, we sometimes fall into one or the other (or both) of a couple of pitfalls. Only one of these pitfalls is particularly well-known: "Buzzwords" -- and even here merely knowing the name doesn't necessarily effect a spontaneous solution. The other deserves more attention: inadequate familiarity with The Relevant Literature.

The key is the notion of what's really relevant. Often, it's the Oral Tradition that matters; published papers, in their attempts to seem scholarly, offer the wrong levels of abstraction or, because of the backgrounds of their authors, are so ill-written as to fail to communicate well. Sometimes, however, that which is truly relevant turns out to be unfindable by a conventional literature searcher because it isn't "in" the field of search.

I wandered into an instructive case in point recently, when it took me over an hour to convince a neophyte to the mysteries of intercomputer networking (who is quite highly regarded in at least one other area of computer science, and is by no means a dummy) that a particular Local Area Network architecture proposal which casually appealed to the notion of "gatewaying" to three or four other networks it didn't have protocols in common with was a Very Bad Thing. "Gateways" is, of course, another one of those bloody buzzwords, and in some contexts it might have been enough just to so label it. But this was a conversation with a bright professional who'd recently been reading up on networks and who wanted really to understand what was so terrible.

So I started by appealing to the Oral Tradition, pointing out that in the ARPA internetworking research community (from which we probably got the term "Gateway" in the first place -- and from which we certainly get the proof of concept for internets) it had been explicitly decided that it would be too hard to deal with connecting autonomous networks whose protocol sets differed "above" the level of Host-to-Communications-Subnetwork-Processor protocol. That is, the kind of Gateway we know how to build -- and, indeed, anything one might call a Gateway -- attaches to two (or more) comm subnets as if it were a Host on each, by appropriately interpreting their respective H-CSNP protocols and doing the right things in hardware (see Figure 1), but for ARPA Internet Gateways each net attached to is assumed to have the same Host-Host Protocol (TCP/IP, in fact

or, anyway, IP and either TCP or some other common-to-both-nets protocol above it), and the same process level protocols (e.g., Telnet, FTP, or whatever). The reason for this assuming of protocol set homogeneity is that they "knew" the alternative was undesirable, because it would involve the translation or mapping between different protocol sets in the Gateways and such T/MG's were obviously to be avoided.

Well, that didn't do the trick. "Why is a T/MG a Bad Thing?" he wanted to know. "Because of the possibility of irreconcilable mismatches in functionality." "For instance?" "Addressing is the most commonly cited." "Addressing?"

Assuming the reader is as bored as I am with the dialogue bit, I'll try to step through some specifics of the sorts of incompatibility one can find between protocol sets in a less theatric manner. Note that the premise of it all is that we don't want to change either pre-existing protocol set. Let's assume for convenience that we are trying to attach just two nets together with a T/MG, and further assume that one of the nets uses the original ARPANET "NCP" -- which consists, strictly speaking, of the unnamed original ARPANET Host-Host Protocol and the unfortunately named "1822", or ARPANET Host-IMP Protocol -- and the other uses TCP/IP.

Host addressing is the most significant problem. NCP-using hosts have "one-dimensional" addresses. That is, there's a field in the Host-IMP "leader" where the Host number goes. When you've assigned all the available values in that field, your net is full until and unless you go back and change all the IMP's and NCP's to deal with a bigger field. Using IP, on the other hand, addresses of Hosts are "two-dimensional". That is, there's an IP header field in which to designate the foreign network and another field in which to designate the foreign Host. (The foregoing is a deliberate oversimplification, by the way.) So if you wanted a Host on an NCP-based net to communicate with a Host on another, TCP-based net you'd have a terrible time of it if you also didn't want to go mucking around inside of all the different NCP implementations, because you don't have a way of expressing the foreign address within your current complement of addressing mechanisms.

There are various tricks available, of course. You could find enough spare bits in the Host-IMP leader or Host-Host header perhaps, and put the needed internet address there. Or you could change the Initial Connection Protocol, or even make the internet address be the first thing transmitted as "data" by the User side of each process-level protocol. The common failing of all such ploys is that you're changing the pre-existing protocols, though, and if

that sort of thing were viewed with equanimity by system proprietors you might as well go the whole hog and change over to the new protocol set across the board. Granted, that's a big jump; but it must be realized that this is just the first of several problems.

(It is the case that you could get around the addressing problem by having the T/MG become more nearly a real Host and terminate the NCP-based side in an application program which would "ask" the user what foreign Host he wants to talk to on the TCP-based side -- at least for Telnet connections. When there's no user around, though, as would be the case in most file transfers, you lose again, unless you fiddle your FTP. In general, this sort of "Janus Host" -- after the Roman deity with two faces, who was according to some sources the god of gateways (!) -- confers extremely limited functionality anyway; but in some practical cases it can be better than trying for full functionality and coming up empty.)

Then there's the question of what to do about RFNM's. That is, NCP's follow the discipline of waiting until the foreign IMP indicates a Ready for Next Message state exists before sending more data on a given logical connection, but if you're talking to a T/MG, its IMP is the one you'll get the RFNM from (the real foreign Host might not even be attached to an IMP). Now, I've actually seen a proposal that suggested solving this problem by altering the T/MG's IMP to withhold RFNM's, but that doesn't make me think it's a viable solution. At the very least, the T/MG is going to have to go in for buffering in a big way (see Figure 2). In a possible worst case, the foreign net might not even let you know your last transmission got through without changing its protocols.

Going beyond the NCP-TCP example, a generic topic fraught with the peril of functionality mismatch is that of the Out-of-Band Signal. (There are some who claim it's also an NCP-TCP problem.) The point is that although "any good Host-Host protocol" should have some means of communicating aside from normal messages "on" logical connections, the mechanizations and indeed the semantics of such Out-of-Band Signals often differ. The fear is that the differences may lead to incompatibilities. For example, in NCP the OOBS is an Interrupt command "on" the control link, whereas in TCP it's an Urgent bit in the header of a message "on" the socket. If you want Urgent to be usable in order to have a "virtual quit button", the semantics of the protocol must make it very clear that Urgent is not merely the sort of thing the NBS/ECMA Host-Host protocol calls "Expedited Data". If, that is, the intent of the mechanism is to cause the associated process/job/task to take special action rather than merely the associated protocol interpreter (which need not be

part of the process), you'd better say so -- and none of the ISO-derived protocols I've seen yet does so. And there's not much a T/MG can do if it gets an NCP Interrupt on a control link, notices a Telnet Interrupt Process control code on the associated socket, and doesn't have anything other than Expediting Data to do with it on its other side. (Expedited Data, it may be noted, bears a striking resemblance to taking an SST across the Atlantic, only to find no one on duty in the Customs shed -- and the door locked from the other side.)

Functionality mismatch is not, of course, limited to Host-Host protocols. Indeed, the following interesting situation was observed at University College London: In their "Terminal Gateway", which translates/maps ARPANET Telnet and "Triple X" (CCITT X.25, X.28, X.29), they were able to get data across, as might be expected, but only one option (echoing), which is rather worse than might be expected. (And the UCL people are quite competent, so the problem almost certainly doesn't have to do with inadequate ingenuity.)

It could be argued that the real problem with Expedite Data and Triple X is that some protocol sets are a lot worse than others. I wouldn't dispute that. But it's still the case, to re-use a Great Network One-liner, that:

> sometimes, when you try to turn an apple into an orange, you get back a lemon.

Nor is the likelihood of encountering irresolvable functionality mismatches the only technical shortcoming of Translating/Mapping Gateways. A somewhat subtle but rather fascinating point arises if we ask what happens when traffic is heavy enough to warrant more than one T/MG between a given pair of protocol-incompatible nets (or even if we'd like to add some reliability, regardless of traffic). What happens, if we think about it a little, is a big problem. Suppose you actually could figure out a way to translate/map between two given sets of protocols. That would mean that for each logical connection you had open, you'd have a wealth of state information about it for each net you were gatewaying. But "you" now stand revealed as a single T/MG -- and your clone next door doesn't have that state information, so any logical connection that started its life with you has to spend its life with you, in a state of perpetual monogamy, as it were. Naturally, this epoxied pair-bonding could perhaps be dealt with by still another new protocol between T/MG's, but it's abundantly clear that there will be no easy analogue to no-fault divorce. That is, to put it less metophorically, it becomes at best extremely complex to do translating/mapping at more

than one T/MG for the same logical connection. As with the broader issue of reconciling given protocol sets at all, doing so at multiple loci of control may or may not turn out to be feasible in practice and certainly will be a delicate and complex design task.

One more NCP/TCP problem: When sending mail on an NCP-based net, the mail (actually, File Transfer) protocol currently only uses the addressee's name, because the Host was determined by the Host-Host Protocol. If you're trying to get mail from an NCP-based net to a TCP-based net, though, you're back in the Host addressing bind already discussed. If you don't want to change NCP (which, after all, is being phased out), you have to do something at the process level. You can, but the "Simple Mail Transfer Protocol" to do it takes 62 pages to specify in ARPANET Request for Comments 788.

If things get that complicated when going from NCP to TCP, where there's a close evolutionary link between the Host-Host protocols, and the process-level protocols are nominally the same, what happens when you want to go from DECNET, or from SNA, or from the as-yet incomplete NBS or ISO protocol sets? There may or may not turn out to be any aspects that no amount of ingenuity can reconcile, but it's abundantly clear that Translating/Mapping Gateways are going to have to be far more powerful systems than IP Gateways (which are what you use if both nets use the same protocol sets above the Host to Comm Subnet Processor protocol). And you're going to need a different T/MG for each pair of protocol sets. And you may have to tinker with CSNP internals.... An analogy to the kids' game of Telephone (or Gossip) comes to mind: How much do you lose each time you whisper to your neighbor who in turn whispers to the next neighbor? What, for that matter, if we transplant the game to the United Nations and have the whisperers be translators who have speakers of different languages on each side?

Other problem areas could be adduced. For example, it's clear that interpreting two protocol sets rather than one would take more time, even if it could be done. Also, it should be noted that the RFNM's Problem generalizes into a concern over resolving Flow Control mismatches for any pair of protocol sets, and could lead to the necessity of having more memory for buffers on the T/MG than on any given Host even for those cases where it's doable in principle. But only one other problem area seems particularly major, and that is the old Moving Target bugaboo: For when any protocol changes, so must all the T/MG's involving it, and as there have already been three versions of SNA, presumably a like number of versions of DECNET, and as there are at least two additional levels which ISO should be acknowledging the existence of, the fear of having to re-do T/MG's should serve as a considerable deterrent to doing them

in the first place. (This apparent contravention of the Padlipsky's Law to the effect that Implemented Protocols Have Barely Finite Inertia Of Rest is explained by a brand-new Padlipsky's Law: To The Technologically Naive, Change Equals Progress; To Vendors, Change Equals Profit.)

At any rate, it's just not clear that a given Translating/ Mapping Gateway can even be built; you have to look very closely at the protocol sets in question to determine even that. It's abundantly clear that if a given one can be built it won't be easy to do (see Figure 3). Yet "system architect" after "system architect", apparently in good faith, toss such things into their block diagrams. Assuming that the architectural issue isn't resolved by a fondness for the Gothic in preference to the more modern view that form should follow function, let's pause briefly to visualize an immense, turreted, crenellated, gargoyled ... microprocessor, and return to the question of why this sort of thing happens.

It's clear that buzzwording is a factor. After all, "system architects" in our context are usually employees of contractors and their real role in life is not to build more stately mansions but to get contracts, so it's not surprising to find appeal to the sort of salesmanship that relies more heavily on fast patter than precision. Another good analogy: I once went to one of the big chain electronics stores in response to an ad for a cassette recorder that "ran on batteries or house current" for $18, only to find that they wanted an additional $9 for the (outboard) AC adaptor. Given the complexities of T/MG's, however, in our case it's more like an $18 recorder and a $36 adaptor.

But is buzzwording all there is? Clearly not, for as mentioned earlier there's also ignorance of the Oral Tradition in play. Whether the ignorance is willful or not is probably better left unexamined, but if we're willing to entertain the notion that it's not all a bait-and-switch job akin to the separately-priced AC adaptor, we see that those who casually propose T/MG's haven't done enough homework as to the real state of the art.

What ever became of that early reference to The Relevant Literature, though? Surely you didn't think I'd never ask. The answers are both implied in the assertion that:

Gateways are Heffalumps

as you'll plainly see once you've been reminded of what Heffalumps are. Dipping into The Relevant Literature, then, let's reproduce the opening of the Heffalumps story:

> One day, when Christopher Robin and Winnie-the-Pooh and Piglet were all talking together, Christopher Robin finished the mouthful he was eating and said carelessly: "I saw a Heffalump today, Piglet."
>
> "What was it doing?" asked Piglet.
>
> "Just lumping along," said Christopher Robin. "I don't think it saw me."
>
> "I saw one once," said Piglet. "At least, I think I did," he said. "Only perhaps it wasn't."
>
> "So did I," said Pooh, wondering what a Heffalump was like.
>
> "You don't often see them," said Christopher Robin carelessly.
>
> "Not now," said Piglet.
>
> "Not at this time of year," said Pooh.
>
> Then they all talked about something else, until it was time for Pooh and Piglet to go home together.

(To satisfy the lazy reader -- who'd actually be better off searching for it in both -- it's from Winnie-the Pooh, not The House at Pooh Corner.)

Pooh, in case you still don't recall, decides to make a Heffalump Trap. (Piglet is sorry he didn't think of it first.) He baits it with a jar of honey, after making sure that it really was honey all the way to the bottom, naturally. In the middle of the night, he goes to the Trap to get what's left of the honey and gets his head stuck in the jar. Along comes Piglet, who sees this strange creature with a jar-like head making frightful noises, and, having known no more than Pooh what Heffalumps really were, assumes that a Heffalump has indeed been Trapped and is duly terrified.

It would probably be too moralistic to wonder how much Christopher Robin actually knew about Heffalumps in the first place. The "Decorator", based on the picture on page 60 of my edition, clearly thinks C.R. thought they were elephants, but I still wonder. At best, though, he knew no more about them than the contractor did about Gateways in the proposal that started this whole tirade off.

NOTE: FIGURE 1. Defining Characteristic of All Flavors of Gateways, FIGURE 2. Gateway and Translating/Mapping Gateway, Approximately to Scale, and FIGURE 3. Respective Internals Schematics, may be obtained by writing to: Mike Padlipsky, MITRE Corporation, P.O. Box 208, Bedford, Massachusetts, 01730, or sending computer mail to Padlipsky@ISIA.