INTERNET NOTEBOOK
SECTION: 2.3.3.14

IEN: 43

Identification

INTERCONNECTION OF VIRTUAL CIRCUITS AND DATAGRAMS.

L. POUZIN

ABSTRACT

This paper offers functional solutions to the problem of terminal-host access, when terminals and hosts are attached to different networks. The case being studied is one network based on CCITT recommendations X3, X25, X28, X29, while the other network is based on a regular architecture of layered protocols, using a datagram transmission service.

INTRODUCTION

For the sake of simplicity, the title of this paper is misleading. Indeed, neither virtual circuits nor datagrams are any useful from a user's point of view. They are just two varieties of transmission services. A practical objective is much broader, i.e. the interconnection of two computer networks, based on different architectures and transmission services :

. C-net architecture is along the lines advocated by most computer network designers (figure 1). It exhibits a hierarchy of independent layers, according to proposals now being studied within ISO.

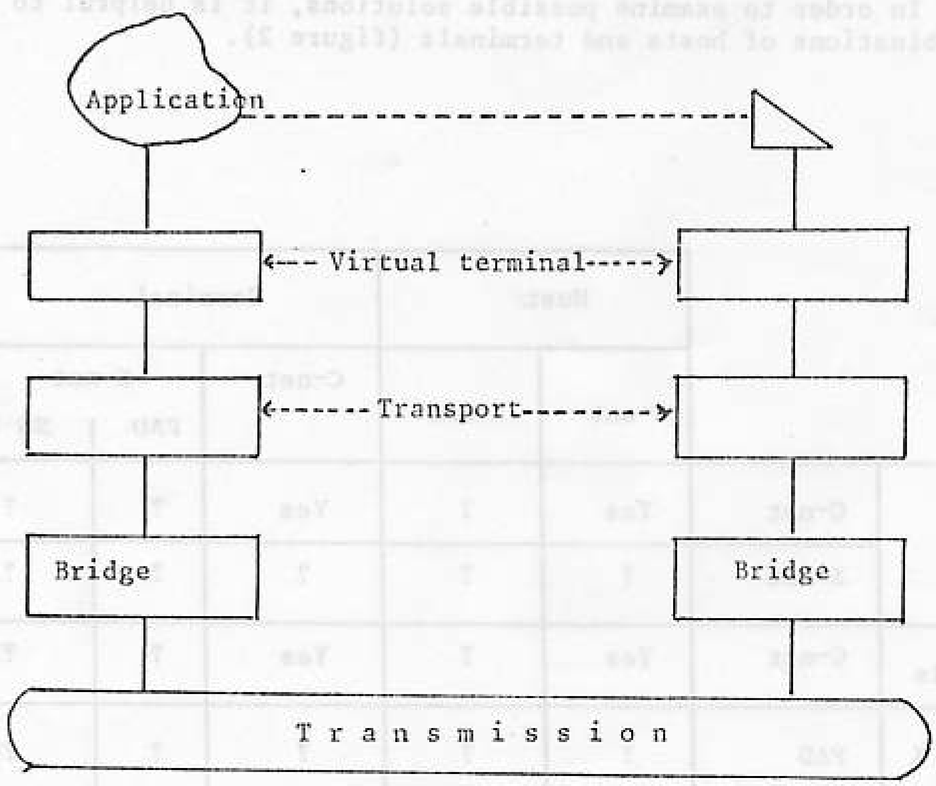

Figure 1 - Computer network architecture

The variety of transmission service is totally immaterial in this model. It could be telephone lines, virtual circuits, or datagrams. A proper adaptation (bridge) must be provided for each variety of transmission service. The simplest one is a telephone line interface. Datagrams require additional packet formatting. Virtual circuits require the most complex bridge in compliance with the CCITT X.25 recommendantion.

The C-net case study uses datagrams.

. X-net is not built yet. Thus, it is premature to rely on a completely defined architecture. The only information given is that the transmissio service is a PTT packet network with PAD's, using X 25, X3, X28, X29. The method for attaching host computers is still undefined. Terminals will likely access PAD's, but some other means will be necessary for the types of terminals not supported by PAD's.

OBJECTIVES

An ideal objective is that any terminal be able to access any data processing service offered on any host, irrespective of which network they are attached to. In order to examine possible solutions, it is helpful to identify various combinations of hosts and terminals (figure 2).

			Host		Terminal		
						X-net	
			C-net	X-net	C-net	PAD	NO PAD
Host		C-net	Yes	?	Yes	?	?
		X-net	?	?	?	?	?
Terminals		C-net	Yes	?	Yes	?	?
	X-net	PAD	?	?	?	?	?
		NO PAD	?	?	?	?	?

Figure 2 - Interworking problems

In the case of C-net, the use of common protocols allows complete inter-working between host services and terminals.

In the case of X-net, the only definite assumption is that hosts use X25, and some terminals use PAD's. This is not sufficient for hosts and terminals to interwork. How this problem will be solved is X-net problem. Therefore, access to X-net hosts from C-net will not be studied further until X-net host protocols are better defined.

In the following, we shall focus on access to C-net hosts from X-net terminals. Access to C-net terminals is left for further study.

X-NET TERMINALS TO C-NET HOSTS

NO-PAD

Since the attachment of terminals to X-net is only constrained to use X25, we make the simplifying assumption that X-net terminals are handled by intelligent terminal controllers (figure 3)

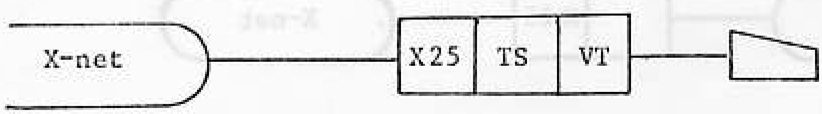

Figure 3 - X-net terminal controller

It is reasonable to assume that controllers will contain protocols used normally in a sound network architecture, i.e. transport station (TS) and virtual terminal (VT). The bridge to X-net is of course X 25.

Let us assume for the moment that TS and VT protocols are compatible in C-net and X-net. Hence, the resulting network architecture is rather familiar, as figure 4 shows. The only gap left is at transmission level.

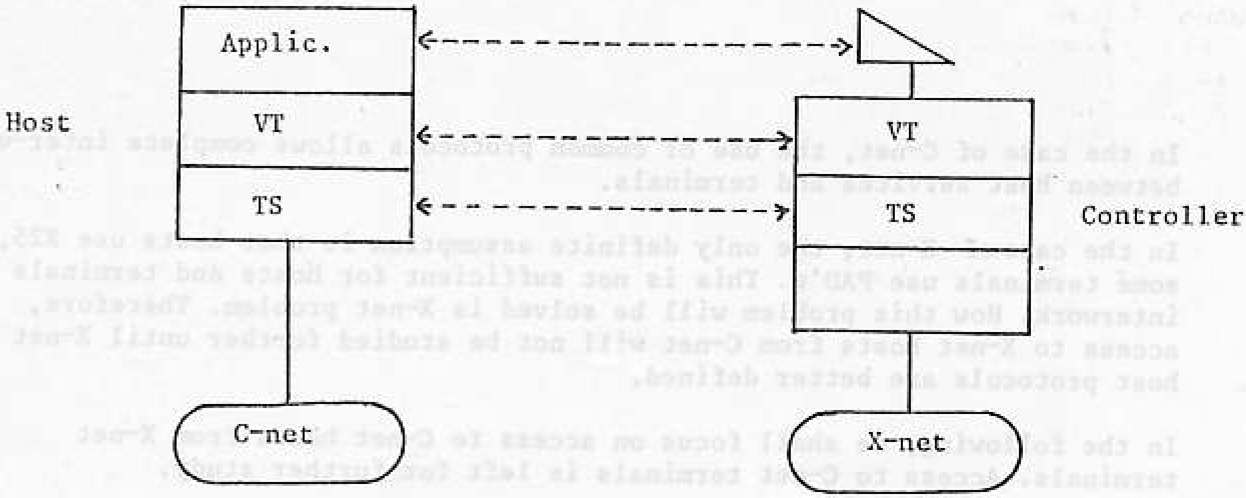

Figure 4 -

As long as packets can find their way between TS's, the whole system works satisfactorily. This can be obtained with a limited additional logic (figure 5)

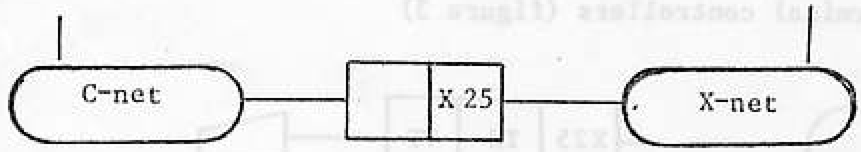

Figure 5 - Simple gateway

A gateway is inserted between C-net and X-net. Its functions are :

. Interface with C-net through a standard ISO line procedure (HDLC)

. Interface with X-net through X25 and several virtual circuits.

. Convert packet formats from datagrams to X25 and vice versa

. Handle signaling packets and abnormal conditions arising on X-net virtual circuits.

Packets going from X-net to C-net are always acceptable, since they can be limited to 255 octets or less as a subscriber option (according to X25).

Packets going from C-net to X-net may have to be fragmented into an X25 packet sequence if the maximum length chosen is less than 255 octets.

Address conversion is not symmetrical. When an X-net terminal connects to a C-net host, it just gives a C-net port number. The X25 bridge in the terminal controller sends the first liaison opening packet to the gateway over a virtual circuit. If no virtual circuit is yet established with the gateway, it opens one. Only one virtual circuit is necessary between each controller and the gateway.From this point on, all packets to C-net are wrapped into X25 data packets addressed to the gateway. There they are unwrapped and forwarded to C-net as datagrams.

Packets originating in C-net are addressed to a port in a controller TS. Since the port address contains the X-net address as seen from C-net, they are delivered to the gateway, and forwarded as data packets over the corresponding virtual circuit. Thus, a table is needed to get the virtual circuit number attached to each controller TS. Since there should only be a small number of TS's in a controller (typically one), the destination address may be shortened by the gateway.

As may be inferred from the above discussion, the major part of the gateway logic boils down to X 25.

If we assume now that TS and/or VT protocols are different in each net, the gateway is an appropriate place to make every necessary conversion. This may or may not be trivial depending on the differences.

· WITH PAD

As opposed to the virtual terminal concept, the PAD does not make terminals appear identical. It is just a remote terminal handler with a slightly simplified command language. Most physical characteristics of the terminal must be taken into account by application programs. Actually, the PAD is no more than an agreement on a way to designate various characteristics and options. By setting parameters to appropriate values, it maps onto a variety of existing terminals. But each terminal takes a different mapping. Thus, a particular PAD setting is nothing but a particular physical terminal.

There is hardly any room left for improvement on the terminal side, since it is attached to the PAD without intermediate logic. But there is more flexibility on the host side, because there is a gateway along the path.

When an X-net terminal connects to a C-net host through an X-net PAD, it may indicate directly the desired port on setting up a virtual call to the gateway. The address given is :

⟨gateway Nr. ⟩ D ⟨ C-net port ⟩

The gateway converts the call packet into a request to its own TS, which opens a liaison with the specified C-net port. The virtual call set up is completed once the liaison has been successfully established. From that point on, packets must be addressed to port numbers when traveling within C-net, and to virtual circuit numbers when traveling within X-net. Thus, the gateway must keep equivalence tables to reformat packets in both directions.

The same considerations as above apply for packet fragmentation.

Some convention should be defined in order to send from the terminal to the gateway a codeword indicating the terminal type. This could be a step within the call set up procedure. From this information, the gateway derives and effects an appropriate PAD setting. The remaining problem is an adaptation between the VT protocol used in C-net, and the PAD seen as a real terminal. Presumably, the number of distinct PAD settings should be kept minimum, so as to avoid an open-ended and excessive variety of adaptations.