UNIVERSITÉ DE MONTRÉAL

Final Report

Study of Standard Issues for Access Protocols of Public Data Networks

DÉPARTEMENT D'INFORMATIQUE

Faculté des arts et des sciences
Université de Montréal
C.P. 6128 Montréal 101

Final Report

Study of Standard Issues for Access Protocols of Public Data Networks

by

G.V. Bochmann
Département d'informatique
Université de Montréal

March 1977

This final report on standard issues and public data networks is respectfully submitted to the Department of Communication as requested and in accordance with contract 02SU.36100-6-0512 between the Department of Communications and the University of Montreal. The work was carried out by G.V. Bochmann (principal investigator) with the collaboration of P. Thibaudeau and P. Goyer under the scientific supervision of L. Durr, C.D. Sheppard, and Y.F. Lum. Appreciation is extended to the many individuals representing the carriers, computer manufacturers and users whose comments provided valuable input to this work.

Table of Content

Introduction

This study of standard issues for access protocols of public data networks concentrates on the following two items:

(a) the interface between start-stop terminals and packet-switched data networks, and

(b) the datagram service.

Our findings on the second item are described in a separate report, entitled "Datagrams as a public packet-switched data transmission service". We refer to that report for any details or conclusions.

As far as item (a) is concerned, our studies gave rise to several reports and working documents that have been submitted for discussion to different groups. Copies of these documents are given in the annexes. We give in the following sections a short description of the background to our work, and the principal conclusions.

The study of the interface between start-stop terminals and packet-switched data networks

We began our work by studying the CCITT documents on the subject, the description of the Telenet interface, and various other proposals for implementing a so-called "virtual terminal" (VT) interface standardizing several aspects of terminal access as seen by the host computer. We also participated at the CCITT Special Rapporteurs Group on Packet Mode Operation held at Geneva in November 1976, a report of which is given in Annex 1. At that time, the plans of the PTT concerning the interface for start-stop terminals (implemented by a packet-assembler-and-deassembler, or PAD) were already well advanced, as reflected in the CCITT document COM VII 54, and some months later a proposal for a draft recommendation (COM VII 62) was presented by the UKPO, France and Telenet carriers. This proposal had been elaborated in close coordination with Bell Canada.

Originally, the question of a character stream interface was one of the points of the proposed study. However, we did not pursue this point any further, since the standard interface proposed by the PTT includes the possibility of having a character stream interface, also called the "transparent mode".

We identified several problems with the approach taken by the PTT. These problems have been explained in the documents "Applying the virtual terminal concept to the interworking between a start-stop DTE and a packet-mode DTE" (see Annex 2) and "Comments on the proposed interface between a PAD and a start-stop DTE" (see Annex 3). The documents were submitted to the NSG VII and the CSA Committee on Data Communications. The discussion in the NSG VII showed that, in the short term, the carriers do not want to change their plans.

The discussion in the CSA Committee showed that there is interest, within CSA and ISO, in the standardization of procedures for communication with interactive terminals. The scope of discussion is larger, including also synchronous and intelligent terminals, and the use of the virtual terminal concept seems appealing. The discussion of these problems is being pursued within the CSA Committee; and for this purpose, we presented a working document on the "Comparison of the virtual terminal approach and the proposal COM VII 62 for the handling of start-stop terminals in packet-switched data networks" (see Annex 4). This document also points out some technical difficulties with the proposal of COM VII 62.

Conclusions

(1) Apart from some technical difficulties, the CCITT proposition COM VII 62 seems to be appropriate for the handling of simple start-stop DTEs, such as teletypes, communicating through a packet-switched data network. However, the interface between the start-stop DTE and the packet-mode DTE is not flexible enough for handling more sophisticated types of start-stop terminals and for adaptation to frame mode DTEs. Therefore, we think that the adoption of COM VII 62 would lead to a proliferation of interfaces, since more flexible and better defined function oriented protocols for terminal access will be needed.

(2) A proposal for minor changes to COM VII 62 is made in section 6 of Annex 4, in view of obtaining a kind of upward compatibility with a virtual terminal protocol. If such changes are made, the proliferation of different incompatible terminal access protocols could possibly be avoided.

(3) The discussion of terminal access protocols within the CSA Committee on Data Communication should be pursued. Canadian contributions to ISO on this subject are envisaged.

(4) The subject of terminal access protocols is only one among many items for which there is growing interest within ISO/TC 97. Other related items are communication systems architecture (for which a new subcommittee is being formed), end-to-end transport protocols and other higher level protocols. A comprehensive overview of these items is given in the last Rapporteur's Report on Project 17 to Subcommittee 6 (N 1417). For information, we have joined this report as Annex 5.

ANNEX 1

REPORT

on the

CCITT Special Rapporteurs Group on Packet Mode Operation

held at

Geneva, 22-26 November 1976

made for the

Department of Communications, Ottawa

by

G.V. Bochmann

1. Introduction

The author of this report attended the Informal Special Rapporteurs Meeting on Packet Mode Operation in the status of an interested party. He attended the meeting in order to advance the work on the research contract "A study of standards issues for access protocols of public data networks", and to report to the Department of Communications on the results of the meeting.

The Special Rapporteurs Group has no decisional power. Its function was to collect a number of contributions on the problems of public data networks, to have an exchange of ideas on the subjects, and to produce propositions that will be discussed at the next Study Group VII meeting in April 1977 (the only meeting of SG VII before 1979). Holding the Special Rapporteurs Group had been considered necessary for obtaining some desired results at the forthcoming Study Group meeting.

2. Topics of discussion and documents involved

In addition to 11 contributions submitted before the meeting (numbered P 1 through P 11), 24 additional contributions were submitted at the meeting (numbered P 12 through P 35). They are listed in Temporary Documents No. 2 and 3. The following topics were covered by the agenda of the meeting:

1. "new Recommendation covering the procedure for interworking between start-stop mode DTE connected through a Public Switched Telephone Network to a Public Data Network (user classes of service 1-2) and packet mode DTE (user classes of service 8-11)" In the following this topic will simply be called PAD (which stands for "packet assembly and deassembly" function). The contributions considered are P1, P5, P12, P21, P26, P28, P31,P33, P35.

2. Revisions for Recommendation X.25. Contributions P4, P5, P11, P14, P23, P24, P25, P30, P32 for levels 1 and 2, and P5, P6, P13, P16, P19, P20, P22, P23, P25, P32, P34 for level 3 (packet level).

3. Interworking of different public data networks. Contributions P7, P8, P27 (from Canada).

4. Datagram facility. Contributions P9, P10, annex to P2, P30, P34.

5. Fast select facility (an alternative replacing the datagram service, as proposed by Japan). Contributions P17, P18.

6. Extensions to X.1, X.2, X.92, X.95 and X.96. Contributions P13, P15, P35.

The rapporteurs R. Despres (France) and J. Wedlake (UK) were together responsible for the topics 1 and 2, and C. Strahlendorf (Canada) was responsible for the remaining topics. C. Strahlendorf chaired also the opening session.

The following editors were appointed to assist the Rapporteurs to record the results of the various activities of the meeting:

a) B. Jamet (France) and C. Martel (Canada) for topic 1 above.

Temporary Documents

No. 25, A: "Introduction"

No. 15, B: "PAD functions and PAD parameters"

No. 16, C: "Description of the DTE/DCE interface for start-stop mode data terminal equipment accessing a public data network offering a packet switched service through the public switched telephone network"

No. 17, D: "Procedures for the exchange of control information and user data between a packet mode DTE and a PAD"

No. 34, E (not completed): "Points for further study"

b) A. Rybczynski (Canada) for topic 2 above.

Temporary Document No. 20: "X.25 - Proposed enhancements, points requiring clarification and points for further study".

c) T. Knappett (UKPO) for topic 3 above.
Temporary Document No. 8: "Elements of a proposed draft Recommendation for an Internetwork Signalling System for International Packet Mode Services".

d) C. Broomfield (UKPO) for topic 4 above.
Temporary Document No. 14: "Possible procedures for the datagram facility".

e) F. Ishino (Japan) for topics 5 and 6 above.
Temporary Document No. 27: "Possible extensions to existing CCITT Recommendations X.1, X.2, X.92, X.95, X.96".

f) G. Kvarnefalk (Sweden) on level 1 of X.25.
Temporary Documents No. 9 and 26: "Proposed revision of level 1 in Recommendation X.25".

g) C.K. Houter (ISO) on level 2 of X.25.
Temporary documents No. 12, 11, 13 and 22: "Editor's report on Recommendation X.25 section 2 (LAP)".

h) W. Raymond (Canada) and S. Ritzenthaler (France) on new definitions.
Temporary documents No. 18 and 30: "List of possible definitions".

i) C.K. Houter (ISO) on Frame Level DTE (sometimes referred to as "simple packet mode DTE" or "single channel packet mode DTE").

Temporary document No. 23: "Frame level DTE".

3. The acting parties at the meeting

About 80 people were at the meeting. The rapporteurs and editors are named above. The main contributions to the discussions during the meeting are shown in the table enclosed. The table shows the written and oral contributions to different topics of discussion made by different parties. We note that the following paragraphs represent the personal opinion of the author.

We think that the reason for holding the Rapporteurs meeting was the need, felt by France, TCTS and UKPO, for finalizing, at the Study Group meeting in April 1977, necessary revisions of Recommendation X.25 and a proposed standard for connecting start-stop terminals to the new data networks (topics 2 and 1). France and UK strongly promoted these issues during the Rapporteurs meeting, whereas TCTS, bound by CTCA, provided a rapporteur and editors to work on the subject. Other administrations, for instance US (except Telnet), Sweden, Japan and Germany, made interventions that stressed the need for a more thorough study of the subject, which would take longer time and therefore delay any agreement to be obtained.

There was a kind of "opposition party" which was represented mainly by ISO, ANSI, IFIP, ECMA, and IBM (France), and which, in our opinion, held a quite sensible view. Their main points were

a) work on topic 1 (PAD) can only proceed in close collaboration with ISO (projects 24 and 17); and the present proposition for the PAD is not sound in a larger context.

b) compatibility of X.25 level 2 with HDLC of ISO.

c) necessary improvements of X.25 level 3.

d) against proliferation of interfaces for terminals; in particular, for a unique HDLC interface for single access DTEs.

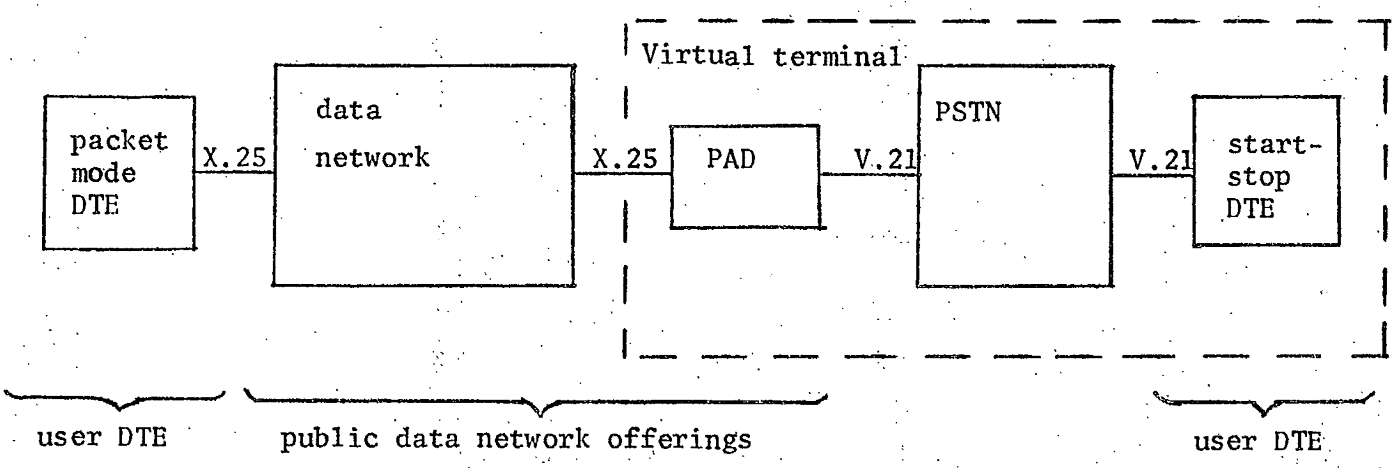
It was generally agreed that the CCITT needs collaboration with ISO. In this context Temporary Document No. 33 points out the next relevant ISO meetings (14-18 March 1977 in Sydney, Australia; 4-5 January 1977 in Utrecht, Netherlands (WG 2 on public data networks)) to which CCITT experts are welcome; and Temporary Documents No. 13 and 28 deal with the present differences between the X.25 level 2 and the ISO HDLC procedures (which are to be eliminated). However, as far as the PAD issue is concerned, it seems that the CCITT is going ahead (independent of the work of ISO) for finding an acceptable recommendation.

4. Results of the discussions

The following sections cover only those subjects of discussion which the author considers most important.

4.1 The PAD (packet assembler and deassembler)

The discussion deals with a PAD for connecting start-stop DTEs through a public switched telephone network (PSTN) to a public data network as shown in the diagram below.



The PAD is a particular kind of "network interface machine" (NIM).

The Temporary Documents 25, 15, 16, 17 and 34 on this subject are, with minor changes, in agreement with contribution P1 (from UKPO) which is basically a stripped-down version of the Telnet "Interactive Terminal Interface Specification".

It is not clear whether the range of PAD parameters provided in the CCITT document is sufficient for handling all envisaged applications. The introduction of additional optional parameters (as contained in the Telnet specifications) may lead to incompatibilities between different networks. Many points are mentioned for further study. We mention among the many detailed points the following more general ones:

a) access from the start-stop DTE to the PAD through different kinds of connections, such as Telex, dedicated line etc.

b) extension of the procedure between the start-stop DTE and the PAD to other transmission rates and codes.

c) applicability of the same procedure between the packet-mode DTE and the PAD for different kinds of network interface machines.

d) the use of the virtual terminal concept, and the extension to more sophisticated interactive terminals.

We note that the topic of the PAD is very closely related to the author's work for DOC. It now seems to be generally agreed that it is desirable to provide a character stream interface ("transparent mode" in Temporary Document No. 15). This simplifies the first point of our work. The second point of our work ("Towards a virtual terminal standard") may have an impact on th PAD standard of the CCITT if it can be completed early enough. On the other hand this work may also be helpful for finding a long-term solution to the PAD question in the necessary collaboration between ISO and CCITT.

The Response ... [illegible] ... at this maximum, with solvent ... [illegible] ... in agreement with conventional ... [illegible]

[illegible]

[illegible]

4.2 Revisions for Recommendation X.25

4.2.1 Access to public data networks (level 1)

In Temporary Documents No. 9 and 26, it is made clear that

a) a dedicated circuit and

b) a circuit switched data transmission service

could be used for a multi-channel access to a public data network. However, there is no agreement whether b) should be offered initially. Some administrations (France, CNCP and possibly others) want to study this question "at home" before making any commitment.

4.2.2 Link access protocol (LAP) (level 2)

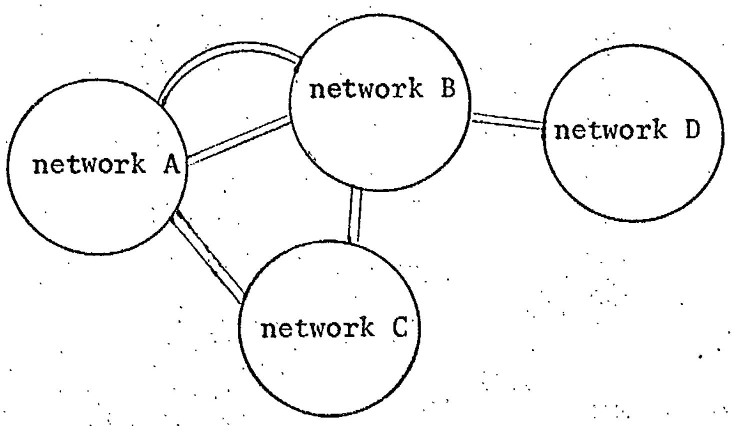
There is a conflict between CCITT and ISO (as mentioned above). It seems that CCITT would eventually include the text of the appropriate ISO specifications in its recommendations.

4.2.3 Virtual call protocol (level 3)

Many questions, amendments and critics were raised on the packet level procedures. They are summarized in Temporary Document No. 20.

4.3 Interworking of different public data networks

Telnet proposed the use of X.25 for building gateways between networks, as shown in the diagram below.



However, some changes to X.25 are necessary. Elements of a proposed draft recommendation are given in Temporary Document No. 8. The addition of accounting information, the use of several physical links for a given gateway (for reliability), and the use of alternative routes for virtual circuits are (among other points) for further study. The issue is generally considered urgent.

4.4 Numbering plan

No contributions were received on this subject. Contributions are sollicited for April 1977 (to be sent directly to Mr. Biehler) considering:

a) a common plan for circuit and packet switched services

b) subaddressing .

The need for subaddressing was stressed by several parties. It has also been found essential for calling a start-stop DTE through a PAD and a public switched telephone network.

4.5 Datagram service

A virtual call "Fast Select" approach has been proposed for those applications that need short exchanges of messages. Favored by some administrations and (probably) more difficult to implement than the (probably less sophisticated) datagram approach, it is not clear, at this time, which of the two approaches is better suited for the intended applications, and how the two approaches compare as far as the difficulties of implementation are concerned.

Possible procedures for a datagram facility and its combination with the X.25 virtual call facility were discussed by an editing group with participants from UKPO, Sweden, Netherlands, ISO and IFIP. It was pointed out that no administration had presently plans for providing a datagram facility. Japan has plans to provide the "Fast Select" facility.

We note that the third point of the author's work for DOC deals with the datagram service. We think that a comparison of the datagram and "Fast select" facilities would be quite useful at the present time.

4.5 Frame level DTE

IBM pointed out the need for a syncronous interface for future terminals. Arguments have been presented for adopting an HDLC protocol for such terminals. This would reduce the number of interfaces to be provided by terminals. But Telnet and France propose the use of X.25 (single channel). This would reduce the number of interfaces to be provided by public data networks. The subject is still quite open, as shown in Temporary Document No. 23.

5. Canadian participation

The Canadian group presented the contribution P 27 entitled "Transit time objectives for an international packet switched public data network". During the discussion within the National Study Group for SG VII this paper was presented by W. Raymond (Teleglobe) and supported by CNCP, but criticized by TCTS. The paper is intended as a starting point for future discussions on the overall delay for international exchange of data. The paper did not entail any discussion at the meeting in Geneva. We believe that this paper, and the subject in general, deserves more attention.

We noted that the Canadian delegates did not express their opinions very often at the meeting in Geneva, although the delegates from TCTS participated strongly in the role of a rapporteur and editors. This is probably caused in part by the fact that they officially represent a Canadian concensus formed in the National Study Group which is not always the opinion of the organization they represent.

Table showing the participation of parties in the discussion of different topics

This table is based on the official minutes and personal notes on the meetings. It is incomplete.

Party		PAD	Revision level 1,2	X.25 level 3	Inter-working	Data-gram	Frame level DTE
Canada	Bell	N			+		
	CNCP						
	Teleglobe				P27		
Germany			P25(*)				
US	gov.	-	P11(I),-			P10(+)	+(NP)
	ATT	P35(*,-)					
France		P12(*), +, N	P14(*),+	+	+	+(FS)	use of X.25
Japan	KDD	P33					
	NTT	P21(C)		P16(*)		P17(FS)	
UK	PO	P1(+),P26(*),N,+	N		P9(+)	P9	
Sweden					-		
Telnet		P28(*)			+		use of X.25
Olivetti						P34(FS)	
IBM France		C,-		P23(-,*)			+
ECMA		P5(-)	P4(I)				P5(NP)
IFIP		P31(-),-		P32(*),-		P2(+)	+(NP)
ISO		-,for collaboration CCITT-ISO		I,-			+(NP)

Explanation:

Px:	contribution
+ :	intervention with positive impact
- :	oppositional intervention
* :	amendments proposed
N :	statement of urgent need
C :	for compatibility with other CCITT standards
I :	for compatibility with ISO HDLC standard
FS:	for Fast Selecting
NP:	no proliferation of terminal interfaces

ANNEX 2

Document prepared for CCITT
Period 1976-1980
SG VII, Packet Switching
Source:

January 1977

Title: Applying the virtual terminal concept to the interworking between a start-stop DTE and a packet mode DTE.

Author: G.V. Bochmann, University of Montreal, Canada

1. Introduction

1.1 General Objectives

We believe that the introduction of a new standard should increase the compatibility of the different equipments that are used together for building an overall system. The PAD takes part of an overall system which typically consists of one or several host computers, a data communication network providing virtual circuits, and a large number of terminals. The PAD's function is the adaptation of the start-stop interface of the terminals to the packet-mode interface of the virtual circuit. A PAD standard should therefore be designed such that it allows for a maximum compatibility between different types of terminals and host computers.

We therefore think that the following objectives are important:

(1) As seen by the packet-mode DTE, the PAD should behave like a standard minimal Virtual Terminal (VT).

The existence of networks makes it possible for a single terminal to access a variety of services provided by host computers. However, the steady increase in special features in terminals makes it difficult to write an application which can communicate with more than a few types of terminals; however, with one level of abstraction many of the variations in terminals disappear.

The definition of a VT protocol represents such an abstraction. It consists of a set of primitive VT functions and some rules for using them, to allow an orderly exchange of information. When a VT protocol is widely accepted as a standard, manufacturers will provide terminals that will conform to that standard, i.e. behave like a VT. As long as this standard terminal is not available, the user still needs a conventional terminal. This terminal plus a terminal handler, like the PAD, will provide an emulation of the VT.

The application need not concern itself with the exact nature of this conventional terminal nor with the local conventions adopted between the handler and this terminal. The application is concerned only with the VT. To make an existing application or service available to the network community, only one adaptation would be needed, the adaptation to the VT.

(2) The PAD should be designed such that it could implement a minimal VT interface for a large variety of different types of physical terminals.

This will simplify the extension of PAD services to most existing and future start-stop terminals.

(3) The minimal VT standard should be designed such that it can be naturally extended to include additional VT functions, which will be required in the near future.

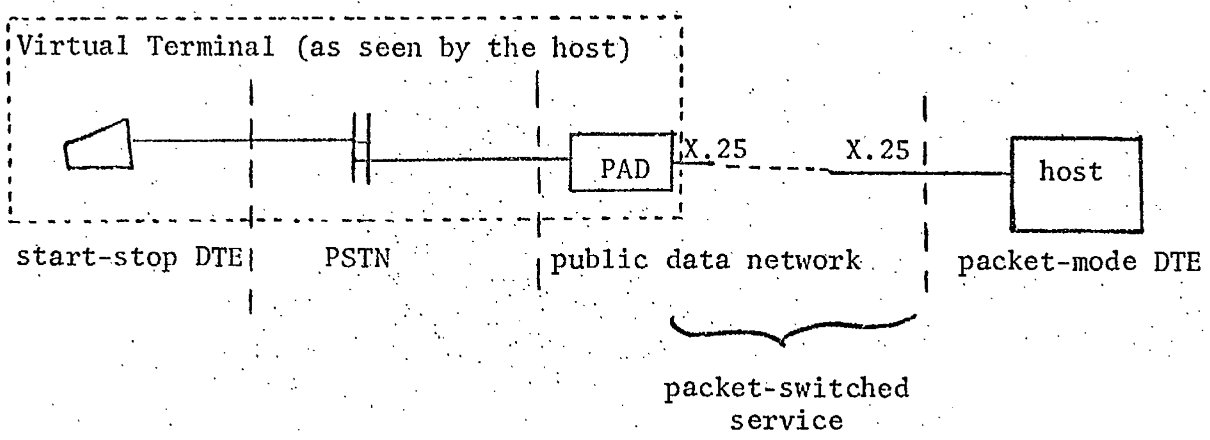


Figure 1: Interworking of a start-stop DTE and a packet-mode DTE through a public data network offering a packet-switched service

We note the following requirements for the VT standard:

(a) The VT protocol must be symmetrical. This allows terminal-terminal or host-host communication just as well as the usual host-terminal communication.

(b) The same protocol should also be used when frame mode DTEs (see COM VII 54, Part III, annex 5) will be connected to virtual circuits. This allows hosts to use the same VT protocol for communication with start-stop and frame level terminals providing the same logical functions.

(c) The VT protocol should not only be applicable in public data networks, but also in private networks. This is important since the VT protocol is an end-to-end protocol, and situations where a terminal in a private network communicates with a terminal connected to a public data network should not be excluded.

Because of the scope of the subject we consider that collaboration with ISO is important.

1.2 <u>Problems with the proposal of Geneva, Nov. 1976</u>
(see COM VII 54 and 62)

We have identified the following problems with the last proposal for interworking between a start-stop DTE and a packet-mode DTE.

1.- As seen by the packet-mode DTE (host), the PAD plays the role of a terminal handler. This implies the following points:

1.1- The host controls the PAD functions.

1.2- Under certain circumstances, the PAD functions can also be controlled by the terminal operator. This may give rise to command collisions and inconsistancies.

1.3- The host has to know certain PAD parameter values (reflecting partly properties of the connected physical terminal) even when the simple mode of operation is adopted, or when all PAD parameters are chosen explicitely by the terminal operator.

1.4- The host-PAD interface is not symmetric.

2.- No effort has been made to standardize the logical functions the start-stop terminal is to perform for the host. A list of such functions (VT functions) could lead to standardization of terminal applications.

3.- Extensions to the PAD functions and parameters are needed to allow for the connection of additional types of physical terminals to the network.

4.- Extensions to the PAD functions and parameters are needed to allow for additional terminal functions (CRT displays in page mode, etc.).

5.- These extensions that will become necessary may not be standardized in time and will lead to incompatibilities between different networks (public as well as private).

6.- These extensions will lead to a very complex interface between the host and the PAD.

1.3 Proposal

Considering the general objectives and the problems with the proposal of Geneva, Nov. 1976 as outlined above, we make the following proposals:

1. To amend the interface between the packet mode DTE and the PAD (COM 54, Part III, Annex 3 and COM 62, Annex 3) to conform with a minimal VT protocol as outlined in sections 2 through 6 of this paper.

   We note that this proposal only affects the data transfer phase, not the call establishment and clearing procedures. The minimal VT interface is very similar, as far as functional capabilities are concerned, with the proposals of COM 54 and 62.

2. To consider for future study the possibility of extending the minimal VT interface to include the additional functions outlined in section 7.

   We note that propositions concerning certain aspects of the interface between the PAD and the start-stop terminal (CMP interface) are discussed in a separate paper.

2. List of functions of the minimal VT

As shown in figure 1, a (symmetrical) VT interface is implemented between the PAD and the packet-mode DTE over a X.25 virtual circuit interface. On the PAD side, it is realized by the PAD together with the physical terminal. The VT functions concern the data transfer phase only. The list of functions of the minimal VT is the following.

2.1 Both way simultaneous data transmission

Data transmitted is structured into logical units, called messages. A message is a sequence of characters representing one or several lines of text. The unit of transmission is generally one message, but could also be smaller, in the extreme case one character.

Data is presented on the presentation unit of the terminal and is structured in (virtual) lines. For virtual lines that are longer than the physical lines of the presentation unit, line folding can be provided.

The VT provides two modes of data transfer (a) standard character set mode and (b) transparent mode. Data transmitted in the standard character set mode consists of characters that belong to a standard character set (displayable characters of Int. Alph. 5), and are coded in a standard code (Int. Alph. 5).

Data transmitted in the transparent mode consists of a sequence of octets which are not interpreted by the PAD, but are transmitted to/from the physical terminal without any change. A facility for the transparent transmission of the BREAK signal is also provided.

2.2 Sharing of the presentation unit between the incoming and outgoing messages

Both, incoming and outgoing messages are displayed on the presentation unit of the terminal (echoing). They are displayed alternatively in the order in which they arrive. In this socalled free running mode the remote DTE, on the other side of the virtual circuit, does not know in which order the messages are presented on the terminal, because of the two way simultaneous mode of message transmission.

The VT also provides a question-response mode which can be used when the remote DTE wants that the message that the terminal operator provides in response to a "question" message from the DTE be identified as such and distinguished from messages the operator may have sent previously. In this case the remote DTE sends a question message, and the first outgoing message from the VT after the receipt of a question message will be identified as a response message.

The remote DTE can send a hide your input indication, on receipt of which the next outgoing message, or the first N characters of it, will not be presented on the presentation unit.

2.3 Attention

The VT may ask the remote DTE for attention. It may also be asked attention.

2.4 Discart data facility

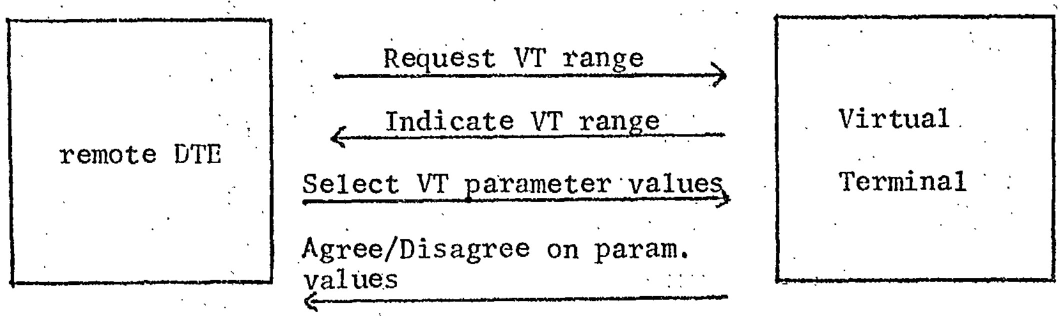
The VT and/or the remote DTE may initiate the discarting of the data messages in transit, separately in each direction of data transfer.

2.5 VT parameter selection

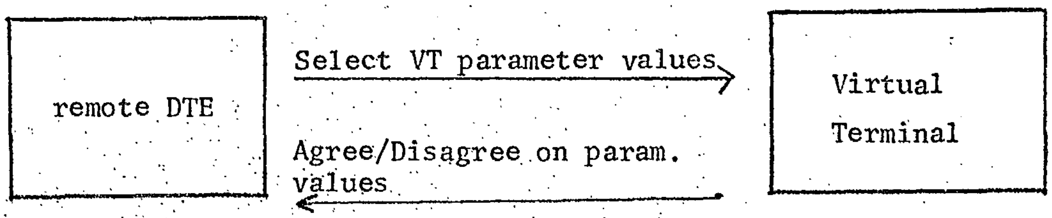
The VT parameters are mainly used for indicating extended VT functions that are provided in addition to the minimal functions outlined in this list. The VT parameter selection facility is necessary when non-minimal VTs are to be used within the network.

The VT protocol provides means for the remote DTE to ask the range of parameters that can be accommodated by the terminal, and set the values of the appropriate parameters, according to the following alternative scenarii:

a) the remote DTE is prepared to adapt to the possibilities of the VT:



b) the application is not prepared to adapt to the capabilities of the terminal, it will directly select the parameter values it needs:



2.6 Error indications

The VT responds by error indications to invalid messages or requests.

2.7 Implementation parameters

The VT may allow the remote DTE to access its local operating parameters (PAD parameters) which reflect the properties of the physical terminal and the mode of operation of the PAD. We note that these parameters are not part of the VT interface, but concern its implementation. Under usual circumstances, the remote DTE will ignore these parameters.

3. The VT end-to-end protocol

3.1 Scope

This section describes the VT protocol as applied between the VT, i.e. the PAD with the connected physical terminal, and the remote DTE. As mentioned above, this protocol is symmetrical, and is concerned with the data transfer phase, not with the call establishment and clearing.

We give in this section a protocol description using an abstract transmission facility, such that the protocol can be adapted to a variety of different real transmission facilities. We suppose that the transmission facility provides for

(a) both way simultaneous transmission of (logical) blocks of data with error control and sequential flow control. Data blocks can be of arbitrary size, and

(b) transmission of interrupts carrying at least one octet of data. The interrupts are transmitted at least as fast as blocks, but independently of the flow control constraints for blocks. We note that the transmission facility will usually divide longer logical blocks into smaller units for transmission.

Section 4 shows how the VT protocol can be adapted to the X.25 virtual circuits used as transmission facility in public data networks.

3.2 The format of transmitted blocks

A block contains a block code and a sequence of items, each representing a primitive function or information for the protocol. The block code is used to distinguish the following kinds of blocks:

(a) data block,

(b) question block,

(c) response block,

each of these blocks contains exactly one VT message;

(d) control block containing control information,

(e) parameter block containing information used for parameter selection,

(f) error block used for error indication.

The format of an item is as follows.

Each item starts with an item code that indicates the primitive being coded in the item. The first bit of the item code indicates whether the item is:

a) One octet long, i.e. the item contains only the item code :

← Item code →
0 X X X X X X X
1 octet

b) Longer than one octet i.e. contains an <u>item parameters</u> field. The indication of the length is contained in a one octet <u>item length</u> field, following the item code.

← Item code →	← Item length →	← Item parameters →
1 X X X X X X X	Value = N	(N-1)octets
1 octet	← - - - - - - N octets - - - - - - - - →	

The value N > 0 in the item length field indicates that the total length of the item is (N+1), i.e. the item parameters fields is (N-1) octets long.

The value 0 in the item length field means that the length is undefined, i.e. the item expands to the end of the block.

<u>Note</u>:

The choice of the item structure, as opposed to other formatting techniques such as the use of control characters, has been made for the following reasons. A big concern in designing a protocol is efficiency, both in transmission <u>and</u> processing. In particular, the "inner loop" (i.e. the part of processing that is most frequently involved) should be considered with great care.

In the VT protocol, the most frequent action to be performed is handling a displayable character, i.e. at the physical terminal, write the character and move the writing head to the next character position. The associated processing in the PAD may be very simple since most real terminals actually use IA5 code for displayable characters. This is not generally true for other functions such as addressing or control functions: some extra processing is generally needed (e.g. new line, padding or code conversion, etc. ...).

Therefore, the block formats must allow efficient distinction between text segments that contain only displayable characters and functions such as addressing or control. This is done by the item structure, the use of which has been generalized to the coding of all primitives.

3.3 Procedures for the VT functions

For each VT function described in Section 2, the following paragraphs describe the procedure followed by the VT.

3.3.1 Data transmission

A complete message is coded in one data block. The following items can be used:

Transmission in standard character set mode:

(a) text item* (variable length): contains a sequence of displayable characters.

(b) new line item (one octet long): indicates the beginning of a new (virtual) line.

(c) text with new line item* (variable length): contains a sequence of displayable characters followed by the beginning of a new line.

Transmission in transparent mode:

(d) transparent data item** (variable length): contains a sequence of octets transmitted without change to/from the physical terminal.

(e) break item (one octet long): can be sent by the VT to indicate a BREAK signal from the physical terminal. When received will generate a BREAK signal to the physical terminal. (This item replaces the "invitation to break" PAD message of the Geneva Nov. 1976 proposal).

* as noted earlier, all characters in this item can be sent to/received from the physical terminal without any checks and processing, provided the physical terminal code is Int. Alph. 5.

** can be sent to/from the physical terminal without any checks and processing.

3.3.2 Hiding and question-response mode

When the VT receives a hide your input item with N as item parameter and the hide option (VT parameter) is ON, the first N characters of the next outgoing message will not be presented on the presentation unit. If N = 0 , the complete message will be hidden.

For question and response messages, the same items are used as for normal messages. Each complete message is transmitted in a corresponding block.

3.3.3 Attention

In order to ask the remote DTE for attention, the VT sends an attention interrupt, and as soon as possible, an attention item in a control block. The attention item may be useful to the remote DTE for determining when, in respect to the sent messages, the VT has sent the attention interrupt.

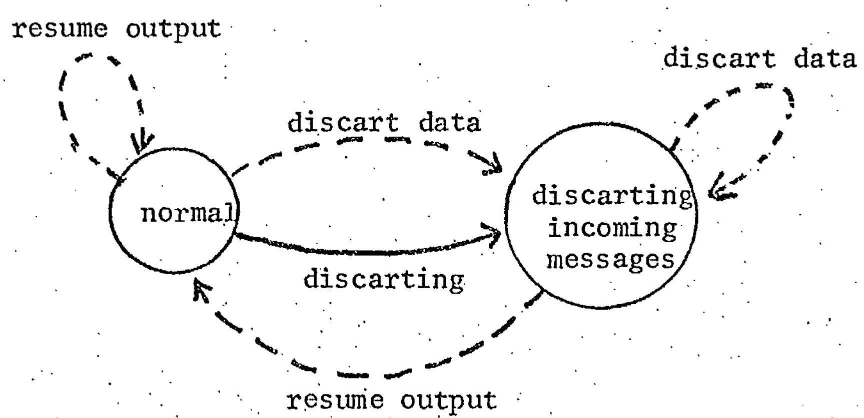
3.3.4 Discart data facility

The procedures for discarting incoming data messages and for having the last outgoing data messages discarted at the remote DTE are shown in the figures 2 (a) and (b). The procedures involve the exchange of discart data and discarting interrupts and the resume output item sent in a control block.

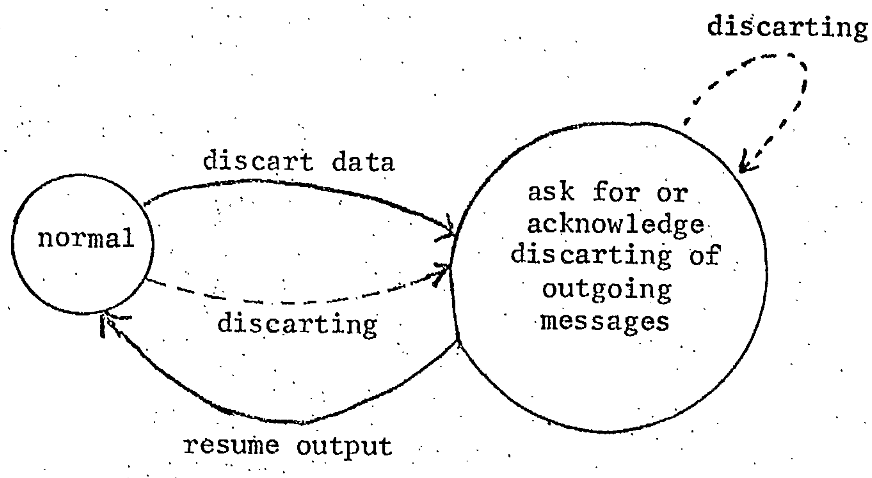
Figure 2: Discart data facility

- - - - -> state transition on reception

———————> sending state transition, on initiative of the VT



(a) Procedure for discarting incoming data messages



(b) Procedure for having outgoing data messages discarted

3.3.5 VT parameter selection

As explained in section 2.5, the procedure involves the items request VT range, indicate VT range, select VT param. values, accept, and refuse, which are sent in a parameter block. The first three kinds of items have a variable length field containing a list of pairs of parameter numbers and parameter ranges, or values respectively. These lists are coded as explained in Document COM VII-35 (P1) section 8.3. The indicate VT range item contains only those asked parameters that have a non-minimal range.

3.3.6 Error indications

In response to the reception of an invalid item, the VT sends an error item in a corresponding block, including as parameter an error code distinguishing between "invalid item code" and "invalid item parameter field", followed by the invalid item received.

In response to the reception of an invalid block code, or in the case of an indicated item length inconsistant with the length of the block, the VT sends an invalid block code or invalid item length item respectively, in an error block.

3.3.7 Implementation parameters

For the selection of implementation parameters (PAD parameters) the same procedure is used as for VT parameters, using the items request PAD range, indicate PAD range, select PAD param. values, accept, and refuse. In addition, the items read PAD parameters and indicate PAD parameters can be used, similarely as outlined in section 8 of Document COM VII-35 (P1).

3.4 Table of primitive items and interrupts used in the VT protocol

Items in data, question, and response blocks:

- text	variable length
- new line	one octet
- text with new line	variable length
- transparent data	variable length
- break	one octet
- hide your input	variable length (three octets)
- error	variable length

Items in control blocks:

- attention	one octet
- resume output	one octet
- error	variable length

Items in parameter blocks:

- request VT range	variable length
- indicate VT range	variable length
- select VT param. values	variable length
- accept	one octet
- refuse	one octet
- request PAD range	variable length
- indicate PAD range	variable length
- select PAD param. values	variable length
- read PAD parameters	variable length
- indicate PAD parameters	variable length
- error	variable length

Items in error blocks:

- invalid block code	variable length
- invalid item length	variable length

Interrupts:

- attention
- discart data
- discarting

3.5 Parameters of the minimal VT

For each parameter, the following indications are given:

- The index indicating the rank of the parameter in the list descriptor
- The description and coding of both, the individual values as well as range of values in the corresponding octet

LINE (Line length)

Index =

Values : 0 indefinite length (line folding is performed)

$0 < n \leq 255$ fixed line length of n positions; $n \leq N$ (virtual lines longer than n will be truncated)

Default value is 0 .

Range : $0 < N \leq 255$; N is the line length on which line folding is based (physical line length).

HIDE (hide your input option)

Index =

Values : 0 no hiding performed

1 hiding performed on request

Default value is 0 .

Range : 0 hide option not available

1 hide option available

4. The VT protocol in public data networks

The VT protocol described in Section 3 is implemented in public data networks by using the virtual circuits of recommendation X.25 as transmission facility. The procedure is as follows:

4.1 Transmission of a block

A block is sent as a complete sequence of data packets (in the sense of X.25) with the more data bit ON in each packet, except in the last one.

4.2 Block codes

A data block is sent as a complete sequence of data packets with the data qualifier OFF in each packet.

All other blocks are sent as a complete sequence of data packets with the data qualifier ON in each packet, where the first octet in the first packet of the sequence is reserved for a block code.

The block code assumes the following values:

kind of the block	value of the block code
error	
question	
response	
control	
parameter	

4.3 Transmission of interrupts

The interrupts are transmitted according to the procedure defined in recommendation X.25.

5. Changes to PAD functions for accommodating the VT protocol

For accommodating the VT protocol as described in Section 3, only small changes are necessary for the PAD functions and the interface to the physical start-stop terminals as described in Com VII 54, Part II, Appendix II A, sections B and C).

5.1 PAD parameters

The possible values for the BREAK parameter (reference 7) should include three bit encodings for

- asking attention
- discarting incoming messages
- discarting outgoing messages .

In the case that physical terminals with a code different from Int. Alph. 5 will be handled, two additional parameters would be needed for specifying possible code conversion:

- indicating the code of the physical terminal
- indicating whether the data received from the physical terminal should be sent in transparent mode or in standard character set mode (possibly code conversion).

5.2 End of message conventions

We mention two possible end-of-message conventions:

(a) the terminal user indicates the end of a message by a particular key stroke. For example a "transmit" key could be used for that purpose. By default, the end of a message could also be associated with the end of a line.

(b) The terminal user is not aware that its data is structured into messages. This could be appropriate when the PAD is in the transparent mode of operation. In this case, each data packet sent could be considered a complete message.

6. Necessary revisions for the text of CCIR Rec. 473 (?)

To be determined (for future study)

7. Proposed future extensions of the VT functions

The following list of functions contains some points for further study to be included in future non-minimal VT specifications. They will comprise an extension of the minimal VT proposal by introducing additional items to the table of Section 3.4.

(b) Transmission of program functions

(c) Additional modes
- Points of presentation attributes in page.
- Mix of different character sets (for mixed alphabet applications)
- Scroll mode with protected fields
- Graphical standards (bit or vector mode)

(d) Cursor addressability
- direct
- relative
- read cursor position
- horizontal and vertical tabulation
- specification of tab stops

(e) Display attributes (colour etc)

(f) Area protecting

6. Necessary revisions for the text of COM VII 62.

To be determined (for future study)

7. Proposed future extensions of the VT functions

The following list of functions contains some points for further study to be included in future non-minimal VT specifications. They will require an extension of the minimal VT protocol by introducing additional items to the table of Section 3.4.

(a) Transmission of program functions

(b) Terminal modes

- format of presentation structured in pages
- choice of different character sets (for multi-alphabet terminals)
- format mode with protected fields
- graphical terminals (dot or vector mode)

(c) Cursor addressing

- direct
- relative
- read cursor position
- horizontal and vertical tabulation
- specification of tab stops

(d) Display enhancement (contrast)

(e) Over-printing

ANNEX 3

Document prepared for January 1977

CCITT
Period 1976-1980

SG VII, Packet Switching

Source:

Title: Comments on the proposed interface between a PAD and a start-stop DTE

Author: G.V. Bochmann and P. Thibaudeau, Univ. of Montreal

1. Introduction

1.1. General

This paper comments on the proposed interface between the PAD and a start-stop terminal (CMP interface) for the interworking of start-stop and packet mode DTEs, as specified in COM VII 54, Part II, Appendix II A, Section B, and COM VII 62.

We feel that the interface should support the evolved characteristics of modern character mode DTEs commonly found on the market. Those modern terminals being so different from each other, the interface to the network should be as flexible as possible in order to adapt to specific characteristics that manufacturers of terminals have implemented (due to the lack of a standard in this field).

The interface proposal mentioned above does not seem to provide a framework that could be naturally extended for handling these different terminal characteristics. Instead the interface is geared at simple terminals of teletypewriter type.

1.2 Proposal

In view of this situation, we propose the following general objectives for the design of the interface between the PAD and the start-stop terminal:

(a) ease of extension for supporting additional (logical) facilities provided by the terminal.

(b) ease of extension for supporting the minimal and additional (logical) facilities on a variety of different real terminals (not necessarily of teletypewriter type).

(c) separation of the terminal handling PAD functions concerning the data transfer phase into

- local functions performed by the PAD and the real terminal for which the remote DTE need not be involved,
- translation between the real terminal and the virtual terminal interface as seen by the remote packet-mode DTE.

The implications of the objectives (a) and (b) are discussed in detail in Sections 2 and 3, respectively. The introduction of the virtual terminal concept is proposed in a separate paper.

2. The handling of terminal functions by the PAD, and the terminal handler command language.

2.1 Terminal functions

We consider, among others, the following terminal functions:

2.1.1 Local functions

- delete last character
- delete current line

2.1.2 Functions related to the minimal VT protocol

- new line
- end of message, if message is built of several lines, or is not terminated by "new line"
- linefold, if transmitted (virtual) line consists of several physical lines of the terminal
- attention
- discart incoming messages (flush output)
- discart outgone messages

2.1.3 Additional (logical) functions (future extensions)

- transmission of program functions
- presentation in page mode
- choice of different character sets
- format mode with protected fields
- direct and relative cursor addressing
- tabulation
- display enhancements
- overprinting

  etc.

2.2. Need for flexibility in the handling of terminal functions

As long as no terminal standards exist, there will be terminals executing similar functions with different procedures; i.e. a given key having a given function may send different characters or character strings if used on different terminals. On the other hand, it may be necessary to send different character strings for different terminals in order to invoke a given function. We feel there is a need for a revision of Int. Alph. 5 in order to include a standard for the terminal functions listed above, as well as the procedure for invoking them. Since such a standard does not exist, and it is improbable that such a standard could be arrived at in the near future, the terminal handling PAD has to recognize and adapt to the differences between existing terminals.

A given terminal function should normally be associated with a key of the terminal, if it exists, or with a sequence of key strokes. Since different real terminals associate different character strings with certain function keys, it is necessary that the terminal handler, too, can associate different character strings with a given terminal function. The set of available functions and the associated character strings depend on the connected terminal type.

We believe that this problem could be solved by providing some standard terminal profiles for those terminal types most commonly used, and allowing the terminal operator to adapt to any particular requirement, if necessary, by changing on line the character strings associated with the different terminal functions.

2.3 Desired characteristics of a command language for terminal handlers.

(a) The command language should be as simple as possible. It should be easy to type and to understand.

(b) Commands should be distinguishable from the rest of the dialogue, for example, by means of a visible escape character.

(c) There could be the possibility of shortening the dialogue by using commands such as

< escape > TERMINAL < standard terminal profile >

2.4 Proposal for a command language

2.4.1 General format

The following format could be used for commands to the terminal handler:

< escape > < command name > [< parameters >] < new line >

The escape character puts the terminal handler into command mode. The command name specifies which function is concerned. The parameter part consists of one or several parameters or could be absent. It typically consists of the character or character string that is to be associated with the function. The new line function terminates the command.

We suggest to use < DLE > as standard escape character. How a command can be distinguished from the rest of the dialogue is for further study.

2.4.2 Some commands

We give the following commands as examples:

< escape > ESC < char.string > < new line >

redefinition of the escape character; simple profile default: < DLE >

< escape > NL < char.string > < new line >

redefinition of new line; simple profile default: < CR >

< escape > EOM < char.string > < new line >

redefinition of end of message; simple profile default: < CR >

< escape > DEL < char.string > < new line >

redefinition of delete last character; simple profile default: < BS >

< escape > KILL < char.string > < new line >

redefinition of delete current line; simple profile default: < CAN >

< escape > FOLD < char.string > < new line >

redefinition of line fold; simple profile default: < LF >

< escape > PAD < integer > < char. string > < new line >

specifies number of padding characters to be inserted after the specified character string; simple profile default: no padding

We note that an empty character string in a command disables the corresponding function.

Possible restrictions on the allowable character strings are for further study.

3. Extensions to the PAD functions and parameters for connecting different types of terminals.

3.1 There is an immediate need for the following functions and parameters.

3.1.1 We propose a speed range that would include the following rates: 110, 134.5, 300, 600, 1200, 2400, 4800 and, possibly, 9600 bits per second. This would satisfy the need of users in almost any application.

The average input rate of a hand-operated terminal is of the order of 1 character each 2 seconds, and the average rate, on output, is of 10 char./sec for low speed and 20 char./sec for high speed terminals. (This difference arises because the user lists more output if it can be done fast). Now, if we consider automatic data transfer on input (in block mode) the average transfer rate on input may increase to 1 or 2 char./sec with transfer of blocks at the peak rate of the terminal (960 char./sec at a bit rate of 9600).

We feel there is an immediate need for supporting these "high speed-automatic transfer" terminals because they tend to be commonly used among different classes of users.

The implementation characteristics (especially hardware) being so different for "low speed-hand operated" and "high speed-automatic transfer" terminals, the consideration of high speed automatic transfer is imperative if we want to avoid that the technical development be frozen by a shortsighted implementation of a proposed standard.

3.1.2 Buffer allocation in the PAD must be sufficient to allow automatic data transfer (as in block or page mode) at the transmission rate of the terminal.

3.1.3 Since the public switched telephone networks usually do not allow high transmission rates (above 300 bps) with reasonably low priced modems facilities should be provided to support private direct lines for connecting start-stop terminals to public data networks.

3.2 The following points should be considered for further study.

3.2.1 Provisions should be made for terminals involving quasi one-way communication, such as credit card readers or printers.

3.2.2 For certain applications or environments, such as banking, it would be of interest to support terminal addressing on multidrop lines.

3.2.3 Terminals with block mode facility having to be triggered to transfer a data block should be triggered by the terminal handler (PAD) in order to release the host system from this function.

ANNEX 4

WORKING DOCUMENT

Title: Comparison of the virtual terminal approach and the proposal COM VII 62 for the handling of start-stop terminals in packet-switched data networks.

Author: G.V. Bochmann, consultant, with the collaboration of P. Thibaudeau

1. Introduction

This paper compares two approaches to the connection of start-stop terminals to packet-switched data networks. One approach is described in CCITT document COM VII 62 which is essentially a refinement of COM VII 35 and the report of the Study Group VII Special Rapporteurs Meeting (Geneva 1976 - COM VII 54), and is submitted to CCITT by UKPO, France and Telenet.

The "virtual terminal" approach is being discussed within different organizations as a possible future standard for the communication with interactive character terminals. Several different proposals for a virtual terminal standard have been made* which all use a basically similar approach. For the discussion of this paper we refer to CSA document 77-12 ("Applying the virtual terminal concept to the interworking between a start-stop DTE and a packet mode DTE" by G.V. Bochmann) which proposes a minimal virtual terminal (MVT) protocol for the communication between the packet-assembler-and-deassembler (PAD) and the remote host.

We note that the virtual terminal approach is mainly concerned with the PAD - host interface during the data transfer phase of the communication, which represents only a small part of the subjects described in COM VII 62.

In the above mentioned document we identified the following objectives and requirements for a virtual terminal standard:

"(1) As seen by the remote packet-mode DTE, the PAD should behave like a standard minimal Virtual Terminal (VT)."

The VT is characterized by the communication functions it can perform. The functions of the MVT and the functions realized by the proposal of COM VII 62 are compared in section 2 of this paper. We conclude that the MVT functions are realized only partly by COM VII 62 and the realization of some of them is not satisfactory.

---

* see for example: P. Schicker, H. Zimmermann, "Proposal for a scroll mode VT in European Informatics Network" (Oct. 1976); P. Schicker, A. Duenki, "VT definition and protocol", Computer Comm. Review, ACM, 6, Oct. 1976; EPSS Liaison Group, "An interactive terminal protocol", INWG General Note # 94, June 1975.

"(2) The PAD should be designed such that it could implement a minimal VT interface for a large variety of different types of physical terminals. This will simplify the extension of PAD services to most existing and future start-stop terminals."

This objective is of secondary importance for this comparison. Provided a VT protocol is used between a host and a PAD, the PAD service for those terminals for which it is not provided by the carriers, could be provided by the user or other groups without introducing any compatibility problems.

"(3) The minimal VT standard should be designed such that it can be naturally extended to include additional VT functions, which will be required in the near future."

This aspect is considered in section 4 of this paper. The comparison shows the advantage of the item sequence codification used for the MVT over the character stream codification used by COM VII 62.

"(a) The VT protocol must be symmetrical. This allows terminal-terminal or host-host communication just as well as the usual host-terminal communication."

This aspect is considered in section 3 of this paper, where some problems for terminal-terminal communications are pointed out for the protocol proposed in COM VII 62.

"(b) The same protocol should also be used when frame level DTEs will be connected to virtual circuits. This allows hosts to use the same VT protocol for communication with start-stop and frame level terminals providing the same logical functions."

"(c) The VT protocol should not only be applicable in public data networks, but also in private networks. This is important since the VT protocol is an end-to-end protocol, and situations where a terminal in a private network communicates with a terminal connected to a public data network should not be excluded."

The last two quoted requirements show the scope of the VT approach. The MVT is proposed as a standard for the communication between a host and an interactive terminal.
The standard is not limited to PADs interfacing with start-stop terminals, but also apply to intelligent terminals that communicate in blocks using line procedures such as X.25 or HDLC in

multi-drop or balanced configurations. Therefore the MVT is defined in such a way that it can easily be adapted to different underlying data transmission facilities. Public data networks with an X.25 interface are only one of these possibilities.

Figure 1 shows the different architectural levels of the communication of an application program in a host computer (a, d) with an intelligent terminal using the MVT, (b) with a PAD and start-stop terminal implementing the MVT, and (c) with a PAD and start-stop terminal as described in COM VII 62. The figure indicates the compatibility between the cases (a) and (b), and the incompatibility with case (c). In section 6 we point out the minimal changes to COM VII 62 for obtaining a kind of upward compatibility with cases (a) and (b).

Section 5 discusses the use of the PAD parameters described in COM VII 62 for the implementation of a MVT interface. Section 4 compares the item sequence codification used by the MVT and the character stream codification used by COM VII 62, and points out some advantages of the former as far as efficiency and simplicity of processing is concerned.

2. Functional comparison

In this section we compare the communication functions proposed for the MVT with the functional capabilities of the PAD as proposed in COM VII 62. For each concept or function defined for the MVT we discuss the corresponding function foreseen in COM VII 62. In those cases where the function is not defined in COM VII 62, it may be obtained by appropriate extensions of the specifications.

2.1 The concept of a message (undefined in COM VII 62)

The concept of a message is mentioned in COM VII 62 using the term "user sequence" in Annex 3, section 1.4. However, it is unclear when the PAD terminates a user sequence to be sent to the remote DTE*, and whether the user at the start-stop terminal can indicate the end of a user sequence.

A possible convention would be that the data forwarding signal also serves as an end of message indicator.

2.2 Virtual lines longer than physical lines (partly implemented in COM VII 62)

For data traffic towards the terminal, the line folding described in COM VII 62 provides a means for outputting virtual lines longer than the physical line length of the terminal. Compared to the MVT, there is however the disadvantage that the remote DTE cannot investigate the physical line length of the terminal when line folding is performed (the corresponding PAD parameter value is zero).

For data traffic towards the remote DTE, COM VII 62 foresees no means for sending (virtual) lines longer than the physical lines of the terminal.

2.3 The "new line" function (incompletely implemented in COM VII 62)

The new line function corresponds to the format effector of COM VII 62, Annex 2, section 3.5.2. However, this format effector is only used for PAD indications, not during the data transfer phase.

During the data transfer phase of COM VII 62, a new line function can be realized by the transmission of the characters CR and LF, although other conventions could be used to the discretion of the start-stop terminal and the remote DTE. Padding characters are inserted by the PAD after each CR sent to the start-stop terminal. But no standard coding is proposed for the new line function.

---

* here and in the following the term "remote DTE" is used for the packet-mode DTE (usually a host computer) which communicates with the PAD through a virtual circuit.

2.4 Standard character set mode (equivalent function in COM VII 62)

2.5 Transparent mode (similar function in COM VII 62, but incomplete)

The MVT distinguishes between text in standard character set mode and text in transparent mode. This distinction is not made in COM VII 62; in fact, it is not necessary as long as the start-stop terminals connected to the PAD always use the standard character set code.

The Break signal cannot be transmitted transparently from the start-stop terminal to the remote DTE, according to COM VII 62. The transmission of an "Indication of Break" PAD message is always coupled with output discarding, since the value 4 is not allowed for the PAD parameter # 7.

2.6 Free running mode (undefined in COM VII 62)

COM VII 62 does not specify in which order characters coming from the remote DTE and characters from the start-stop DTE to be echoed will be displayed on the start-stop DTE. Intermixing of incoming and outgoing text would be permitted. We note that the free running mode of the MVT is based on the message concept.

2.7 Question response mode (non-existant in COM VII 62)

2.8 Input hiding (similar function in COM VII 62, but problematic realization)

Although not meant for this purpose (see COM VII 54, Part III, annex 2, 3,2) the echo PAD parameter defined in COM VII 62 could be used for obtaining a (questionable) implementation of input hiding. The remote DTE can have the input on the start-stop terminal hidden, by sending a "set parameter" PAD message (Annex 3, section 4.4) setting the echo PAD parameter (reference # 2) to the value 0 (no echo), and resetting the value to 1 (echo active) afterwards. This mecanism gives rise to the following unresolved problems:

(1) The mecanism only works when the echoing is done by the PAD. It is inapplicable when the start-stop terminal or its modem do the echoing. (We note that this possibility is foreseen by the "hide" parameter of the MVT).

(2) The user of the start-stop terminal may change the value of the echo PAD parameter, using the PAD command signal described in Annex 2, section 3.3.2, without the remote DTE noticing.

(3) Irregular packet delays in the network may give rise to the following problems:

(a) If the PAD message setting the echo off is sent by the remote DTE after a packet containing the text, say "type in your password, please", and the packet containing the PAD message is delayed in the network, the user may type his password before the echo is switched off.

(b) If, on the contrary, the PAD message is sent before the text packet, and again the second packet, containing the text, is delayed, it may happen that a user who is inputting data wonders why the system stops echoing.

We note that according to COM VII 62, the echo of the PAD can be switched off completely. This option is not foreseen for the MVT, but can be implemented in a VT PAD as explained in section 5 of this paper.

2.9 Attention (similar function in COM VII 62)

The sending of an "Interrupt" (see COM VII 62, Annex 2, section 4.9. 2.4) could be interpreted as an attention signal.

The MVT protocol provides, in addition to an attention interrupt sent to the remote DTE, an attention item that travels in sequence with the data and may be used by the remote DTE to determine at which instant the attention signal was initiated by the VT.

As implemented in COM VII 62, the attention interrupt is in possible conflict with the discard output facility, as explained below.

2.10 Discard data facility (different implementation in COM VII 62, partly undefined, problematic)

As explained in COM VII 62, Annex 1, section 3.7 and 3.8, and Annex 3, section 3.3, the discarding of incoming data (discard output) is controlled by the PAD parameter # 8. The logic of the discard output protocol of COM VII 62 is similar to the protocol of the MVT for discarding incoming messages (see MVT, section 3.3.4), however, the coding of the exchanged information is different.

In the case of COM VII 62, it is not clear whether the remote DTE, after receiving an interrupt packet, should discard the received data packets until the 'Indication of Break' PAD message sent by the PAD (corresponding to the "resume output" in the MVT), is encountered. If it does not then COM VII provides no means for having the last outgone messages, sent by the PAD, discarded at the host (as is possible for MVT). If it does then the discarding of data is always done simultaneously in both directions (the MVT allows discarding independently in both directions), and the following conflict occurs:

> The reception, by the remote DTE, indicates that received messages should be discarded until an Indication of Break is encountered. This action will also be followed when an attention interrupt (see section 2.9 above) is received. In the latter case, however, no Indication of Break is likely to follow.

A solution to this problem would be the distinction, in COM VII 62, of several different kinds of interrupt packets, using different data field values (see COM VII 62, Annex 3, section 1.5.5).

2.11 VT parameter selection (there are no VT parameters in the approach of COM VII 62

2.12 Implementation parameters (very similar to COM VII 62)

The PAD parameters foreseen in COM VII 62 could be used for the implementation of a MVT interface with the remote DTE, as explained in section 5 of this paper.

The MVT foresees control messages for requesting and indicating the range of possible values of PAD parameters. This facility is not foreseen in COM VII 62.

3. Some remarks on symmetry

The protocol between the remote DTE and the PAD, according to COM VII 62, is not completely symmetrical. Symmetry of this protocol is desirable, because in the case of a symmetrical protocol, this same protocol could be used not only for the communication between a start-stop terminal and a host, but also for terminal-terminal and host-host communication.

First of all we note that COM VII 62 does not foresee the possibility of a PAD accepting an incoming virtual call to a start-stop terminal. Without this possibility, terminal-terminal communication is not possible through a public data network. This possibility is however for further study.

As far as the data transfer phase of COM VII 62 is concerned, the protocol is symmetrical, except for the handling of the PAD parameters. The non-symmetrical handling of PAD parameters can give rise to problems such as the following. Suppose that two start-stop terminals communicate via two PAD connected via a virtual circuit. Suppose one of the terminal initiates the discarding of output by using the break signal (with the appropriate parameter values being set in the PAD) and waits for the reception of the "resume output" message to be sent by the remote DTE. It will wait forever, since the remote PAD does not respond to the received interrupt packet.

It seems to be reasonable that the functions of a MVT should be implemented on each (virtual) terminal at least as far as the response to incoming data and control items is concerned, but not necessarily as far as the initiation of the functions is concerned. For example, considering the "hide input" function, it seems reasonable that an interactive terminal be able to respond to an incoming "hide your input" item, but not necessarily be able to send such an item. A typical MVT would have the capabilities shown in the following table:

function	capability for initiation	capability for response
text transmission (standard code)	X	X
text transmission (transparent)		X
new line	X	X
question-response		X
hide input		X
attention	X	X
discarding output (incoming data)	X	X*
discarding outgone data		X
parameter selection		X

* Lacking of this capability in COM VII 62 leads to the problem mentioned above.

4. Formats of codification

4.1 The character stream codification

According to COM VII 62, all information exchanged between the PAD and the remote DTE, except the PAD messages, is coded as a stream of characters of the International Alphabet # 5. Most characters represent text to be displayed, but other functions are also coded as character sequences. For example, the new line function would usually be represented by the two characters CR and LF. We call this approach the "character stream codification".

If the character stream approach is to be retained for the realization of a VT protocol, it has to be extended. Each function of the VT requires one or several codes which must be defined as special control characters or sequence of characters. This approach has been taken by the manufactures of more sophisticated start-stop terminals without waiting for a standard, which lead to the present incompatibilities.

The PAD or the host that receive character stream encoded information have to decode this information by an algorithm similar to the following one:

Decoding algorithm for character stream information

```
for each received character  C  do

begin
     test bits # 7 and # 6
     if  C  is a control character

     then begin

          look up  C  in a table of control characters
          if  C  is the first character of a function code (i.e. CR or DLE)
          then decode the function and execute it
          else transmit  C
          end

     else begin

          (possibly) test whether  C  is DEL or some displayable
               character used as the first character of a
               function code

          if so

          then decode the function and execute it

          else transmit  C

          end

end.
```

4.2 The item sequence codification

According to the MVT, all information exchanged between the VT (or PAD) and the remote DTE is coded as a sequence of items, where each item contains a one-octet item code, a (possibly implicit) length indication, and possibly item parameter values. Each function of the VT is coded as one or several items. For example, a line of displayable text could be coded as a text item followed by a new line item

text code	length	char.1	char.2	---	char.n	NL code

or by a single text with new line item (where we assume implicit length indication, as explained below)

	char.1	char.2	---	char.n

text with new line code

The PAD or the host that receive item encoded information have to decode this information by an algorithm similar to the following one:

Decoding algorithm for item sequence information

```
for each received item do
begin
     if bit # 8 of code = 0
     then length := 1
     else if bit # 7 of code = 0
          then length := up to the end of the block
          else length := value of next octet;
     depending on the value of the item code, goto the corresponding
          function routine
end.
```

Algorithm of the text function routine

```
for  i = 1  through length do
          transmit next character in the parameter field.
```

4.3 Implicit item length indication

The MVT protocol provides for the following three indications of the item length:

(a) items without parameters (1 octet long) have an item code of the form

0XXXXXXX

(b) items with parameters

(b_1) the item length is indicated by the value of the second octet (item length) of the item.

(b_2) the parameter field extends to the end of the block.

We call the case (b_2) implicit item length indication.

The proposed MVT protocol codes the implicit item length indication by an item length octet value equal to zero. We suggest now a different coding, using a second bit of the item code for distinguishing between the cases (b_1) and (b_2). Therefore the length field octet is only present in case (b_1). For example, the item codes could be of the following form

case (b_1) :	11XXXXXX	length	parameters ---
case (b_2) :	10XXXXXX	parameters ---	

↑ end of block

4.4 Comparison of the character stream and item sequence codifications

4.4.1 The VT interface

For the interface between the VT (or PAD) and the remote DTE, the item sequence codification presents the following advantages over the character stream codification:

(a) The processing (i.e. decoding algorithm) is simpler and more efficient.

The relative simplicity of the processing is obvious from a comparison of the two algorithms above. We note that it is a well established practice of software engineering to structure data in such a form that the processing becomes simpler. The structuring into items of the information exchanged in the considered protocol is just another example of this useful programming technique.

The higher efficiency comes from the fact that the decoding algorithm for the character stream information has to test each character to see whether it introduces a special function. For example, according to COM VII 62, the PAD has to test each character sent by the host to see whether it is a CR, in which case it has to insert the padding characters. For VTs, the corresponding test becomes more complex, such as shown in the decoding algorithm above. In the case of item codification, however, the decoding of the item code is only done once for a whole line of text, and the characters of the line are transmitted (i.e. displayed) without any further tests.

(b) Extensions for coding additional functions are straightforward. It is sufficient to define new values for the item code.

We note that, due to the advantage (a), the item codification may also be a good candidate for coding formated text to be stored on files, or for manipulation by text editors and other programs for text processing. However, this aspect is outside the scope of data communication, and should be studied by the appropriate standard organization.

4.4.2 The interface to the start-stop terminal

For the interface between the PAD and the start-stop terminal the character stream codification has to be used. The item codification is not compatible with the available start-stop terminals. There is however the problem that for most functions to be considered for a VT interface, incompatible character stream codifications have already been adopted for different start-stop terminals. We note that the inefficiency of the decoding algorithm has a limited impact because the data traffic is usually much slower in the direction from the start-stop terminal to the PAD as compared to from the remote DTE.

As defined in COM VII 62, the PAD has to identify the following characters (received from the start-stop terminal) which introduce a function:

CR :	padding
DLE :	signal to escape from Data Transfer state
Break signal:	used for many different functions, such as virtual circuit reset, escape signal (alternatively to DLE), discard output, attention (i.e. interrupt).

In the case of a MVT or a more sophisticated VT implemented by a start-stop terminal and a PAD, the PAD has to identify additional control characters which are associated with the VT functions. The exact format of the interface to the start-stop terminal could be determined following the ideas outlined in COM VII 62, Annex 2 and CSA DC 77-13 ("Comments on the proposed interface between the PAD and a start-stop DTE" by G.V. Bochmann and P. Thibaudeau).

5. The role of PAD parameters in a VT implementation

In the case that a MVT is implemented by a PAD and a start-stop terminal, the PAD would usually contain some socalled PAD parameters that indicate (a) how the VT interface is translated into the start-stop interface, and (b) the state of the VT protocol. In this section, we consider the PAD parameters described in COM VII 62 and discuss some changes and additions for accomodating a VT protocol.

5.1 Changes for implementing the MVT with a "simple" start-stop terminal

As given in detail below, it is sufficient to change slightly the definition of 4 PAD parameters among the 11 parameters defined in COM VII 62 in order to obtain an implementation of a MVT with a "simple" start-stop terminal. By a "simple" terminal we mean a terminal that can be handled by a PAD according to COM VII 62 (i.e. a 100 or 300 bps teletype kind of terminal).

Parameter # 2: Echo

If the VT parameter HIDE (hide your input option) is ON then PAD parameter # 2 should be read-only.

Parameter # 7: Selection of the procedure on Break from the start-stop mode DTE

(a) The value 4 should be allowed (for the transparent transmission of the Break signal).

(b) Additional values should be defined for possibly associating the functions "attention", "discard incoming data", and "discard outgone data" with the Break signal. We note, however, that it would be useful to be able to associate these functions also with certain other keys of the start-stop mode DTE.

Parameter # 8: Discard output

Should be a read-only parameter.

Parameter # 10: Line folding

Should be a read-only parameter.

We note that the values of the read-only parameters would usually be changed by the PAD in accordance with the exchange of information through the MVT protocol between the PAD and the remote DTE.

5.2 Additional PAD parameters

For implementing a MVT with a start-stop terminal that uses a non-standard character code, the following additional parameters are needed:

(a) code of the start-stop terminal

(b) code conversion to be performed (yes or no).

For the implementation of a MVT with different ("non-simple") types of start-stop terminals, and for the implementation of a flexible command language as proposed in CSA DC 77-13 ("Comments on the proposed interface between the PAD and a start-stop DTE"), there is a need for the possibility of additional values of PAD parameters and for the introduction of some new PAD parameters. Similarely, extensions to the PAD parameters are necessary for the implementation of more sophisticated VTs.

6. Minimal changes to COM VII 62 for obtaining "upward compatibility" with a VT protocol

Considering that public data networks will probably support in the near future a PAD service similar to the specification of COM VII 62, and ISO will probably define a standard for communication with interactive character terminals somehow along the lines of the MVT, we believe that it would be very useful to the carriers as well as the data processing community if "upward compatibility" could be obtained between the PAD standard adopted by the CCITT and the VT standard elaborated by ISO. By "upward compatibility" we mean that an application which has been programmed for communicating with a start-stop terminal through a carrier provided PAD (as described in COM VII 62) should also be able to communicate, without necessitating any change, with a MVT through a MVT protocol implemented by a PAD and the same type of start-stop terminal (we therefore ignore the incompatibilities in the handling of new line and transparent text transmission; this is only partial upward compatibility).

The next two subsections specify the minimal changes to COM VII 62 for obtaining this kind of upward compatibility.

6.1 Incompatibility of functions provided

From the detailed comparison in section 2 we conclude that the functions of input hiding and output discarding are not satisfactorily implemented in COM VII 62. These functions should either not be provided, or be provided according to the MVT specifications.

The handling of the Break signal from the start-stop terminal also needs some reconsideration.

6.2 Incompatibility of coding

Considering the "character stream codification" of COM VII 62, the "item sequence codification" of the MVT, and the comparison of section 4, the following changes of codification should be made to packets received and sent by the PAD from/to the remote DTE:

(a) Insert one octet, representing the text item code with implicit length indication, at the beginning of each user sequence.

(b) Replace the "Indication of Break" PAD message by a user sequence (where the data qualifier is equal to zero) consisting of one octet, representing a break item code.

(c) Insert one octet, representing the parameter block code, at the beginning of each PAD message.

(d) Obtain agreement between the PAD message codes, which act as parameter item codes, and the coding scheme described in section 4.3 (case b_2).

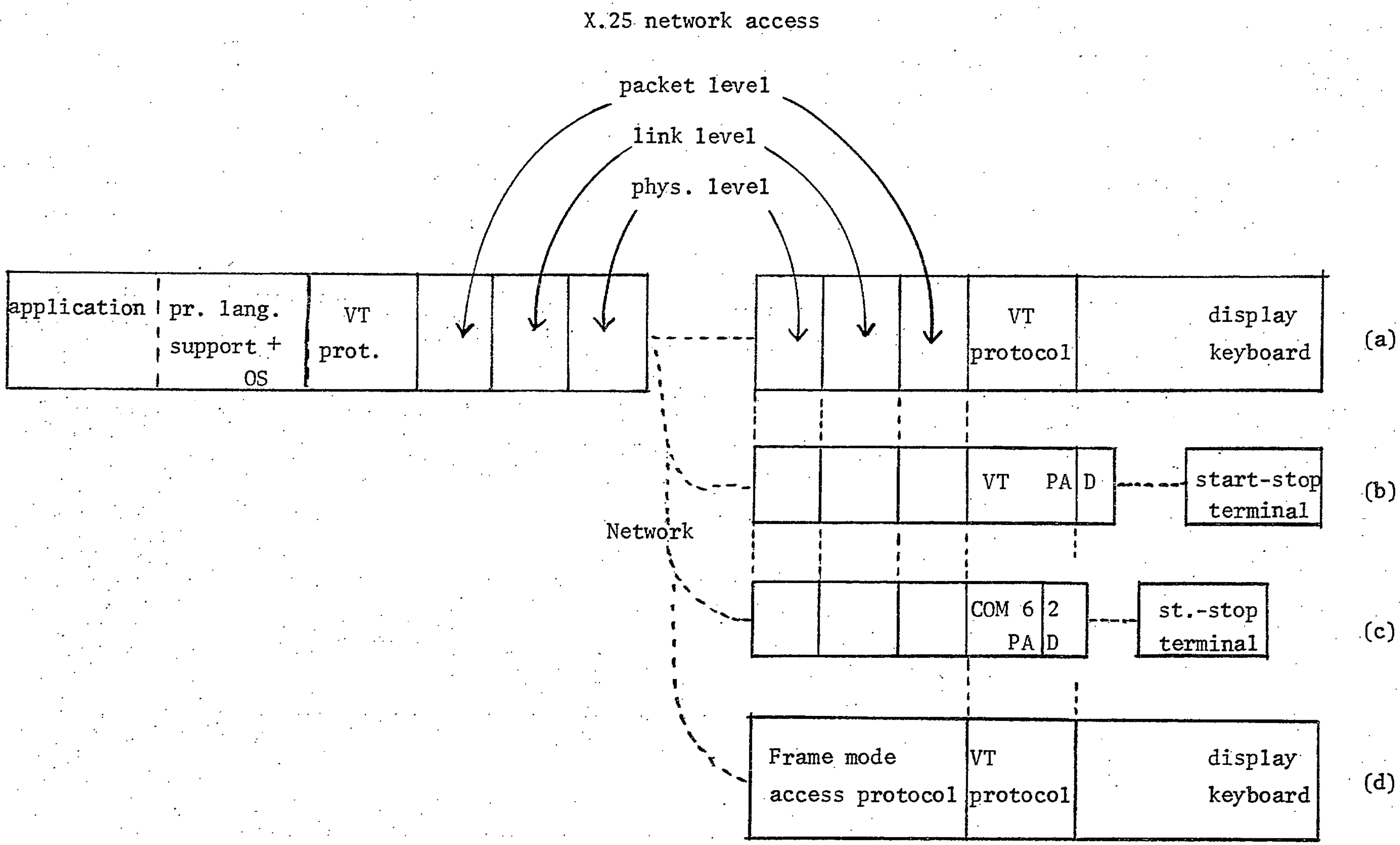


Figure 1: Architectural levels of protocols for communication with an interactive terminal
(a) Intelligent terminal using X.25 access procedure
(b) VT PAD with start-stop terminal
(c) COM VII 62 PAD with start-stop terminal
(d) Intelligent terminal (of kind "Frame Mode")

ANNEX 5

Association Française
de Normalisation
Tour EUROPE
CEDEX 7
92 080 PARIS LA DEFENSE
Tél. 788. 11. 11.

I S O

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION

ISO/TC 97/SC 6 N 1417

ISO/TC 97/SC 6
DATA COMMUNICATIONS
Secretariat :
ANSI (USA)

February 1977

RAPPORTEUR REPORT ON PROJECT 17

COMMUNICATION SYSTEMS ARCHITECTURE

AND END TO END PROTOCOLS

Hubert ZIMMERMANN
Rapporteur, Project 17.

1 - INTRODUCTION

In accordance with Resolution 6 of N 1258, this report from the Rapporteur on Project 17, Communication Systems Architecture and End-to-End Protocols, is submitted to ISO/TC 97/ SC.6.

The documents considered in this report are listed in Appendix A, and for the sake of brevity, the ISO/TC 97/SC 6 document number rather than the paper title will be referred to in the main body of this report.

Contributions on Project 17 cover various aspects of standardization in "Communication Systems Architecture and End-to-End Protocols" and each section of this report discusses one aspect of Project 17 :

- Need for standards in high level protocols (section 2)
- Need for a reference architecture (section 3)
- Architecture model (section 4)
- Layering of functions (section 5)
- Points for study (section 6)

2 - NEED FOR STANDARDS IN HIGH LEVEL PROTOCOLS

Several contributions put an emphasis on the urgent need for standards in high level protocols to allow interworking between equipment connected to the same data transmission network (N 1167, 1173, 1258, 1280, 1296, 1369). It is stated (N 1350) that public data transmission networks will favour "open working" where data processing equipments will have to be prepared to communicate with any other data processing equipment, i.e. by means of standard end-to-end protocols.

3 - NEED FOR A REFERENCE ARCHITECTURE

The need for a Communication Systems Architecture has been raised for several years within SC 6 (e.g. N 771, 798). More recent contributions state that a reference architecture is needed as a general basis for future work in standardization of network protocols, including end-to-end (also called high level) protocols (N 1173, 1280, 1350, 1356, 1367). The definition of a general architecture for communications system, including public data transmission networks as well as networks constructed by the user for its own use, is also being viewed (N 1356) as a possible means for coordinating the work being done in ISO and in CCITT (e.g. PADs and private concentrators should not look different to the remote computer).

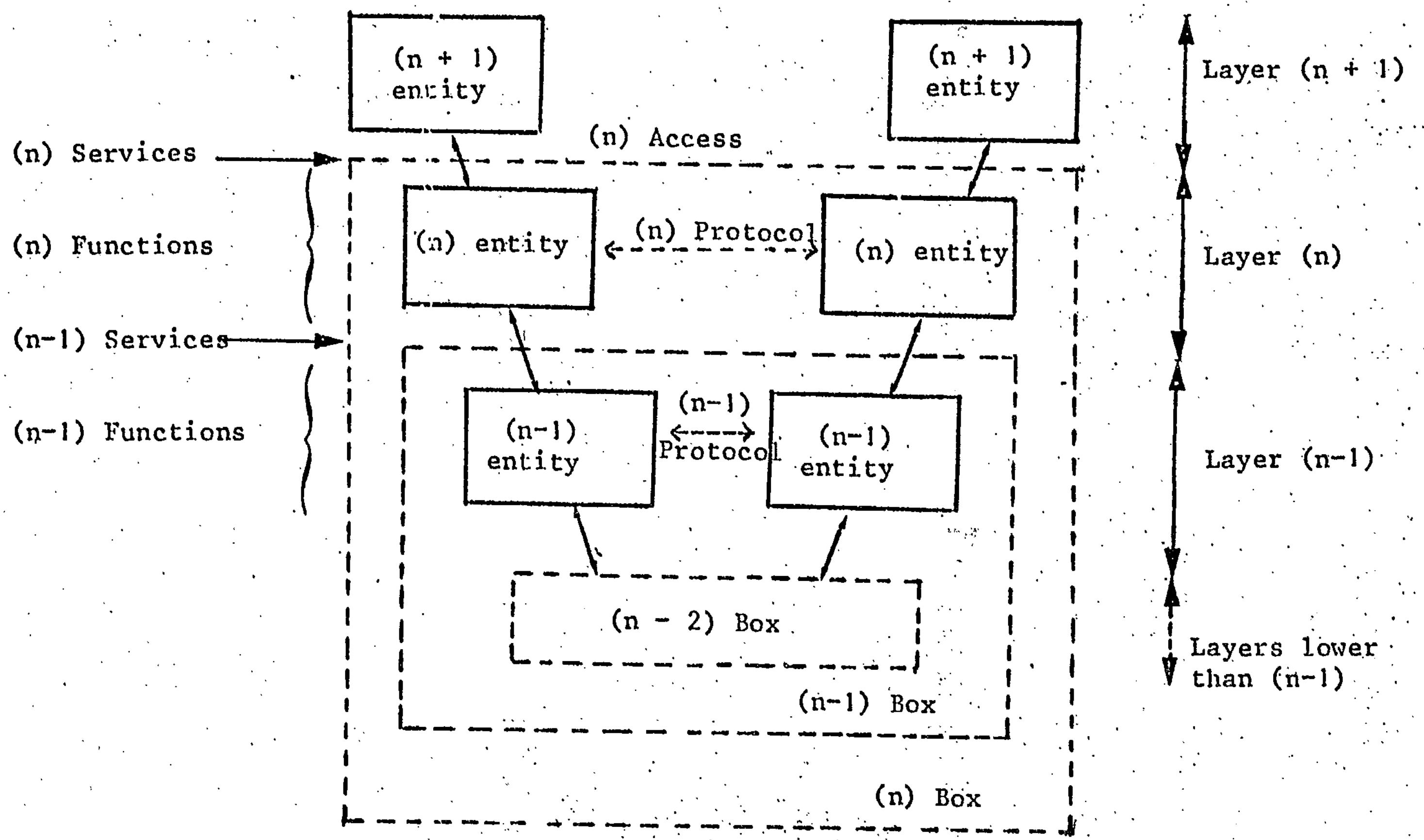
It is also considered (N 1350) that Architecture will help in assigning responsibilities for standardization within the various groups involved, and pointing out the requirements for links and coordination between them.

4 - ARCHITECTURE MODEL

There is a general agreement that a layered architecture be used to describe informatic networks and standardize protocols. Several contributions also insist on the notion of structured layering, i.e. on the fact that higher layers should not be aware of the internal structure of lower layers, they should consider these lower layers as a black box (N 771, 921, 1167, 1248, 1259, 1283, 1286, 1296, 1369). It is being put forward that looking at pieces of a network as black boxes provides the flexibility required for implementation and will allow evolution of subsets of protocols to follow technical and/or technological changes. In addition, structured layering permits to split standardization responsibilities with maximum freedom to each party.

A comprehensive definition of a model to describe a layered architecture is proposed in N 1369. This definition is consistent with the views expressed in all contributions on that aspect of project 17. This definition is as follows :

"The network is logically made of successive layers. The bottom layers correspond to telecommunication means and the upper layers correspond to applications and users.



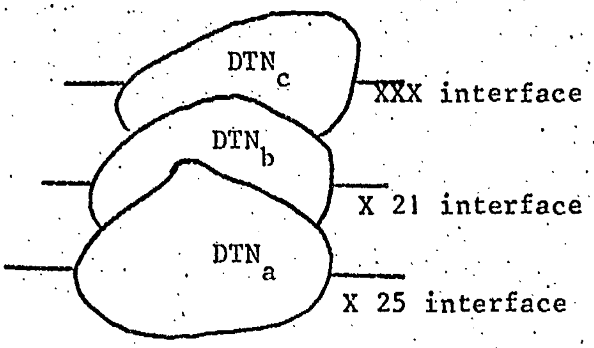
- Layer (n) of the structures makes use of (n-1) Services provided by the lower layers through the (n-1) Access.
- The structure of these lower layers is not known by layer (n) that considers only the services provided by an (n-1) Box.
- Layer (n) is made of (n) Entities that cooperate according to an (n) Protocol.
- The (n) Entities perform (n) Functions using the (n-1) Services, to provide (n) Services to the Layer (n+1).
- The specification of a layer of the architecture must in some way refer to the set of services provided by lower layers. This will be done by making use of Access-Functions. The set of access functions to a service should be viewed simply as a means to describe the logical structure of the network. It does not necessarily imply the existence of the corresponding interface in any implementation of a piece of the network".

It is the rapporteur's opinion that this definition can be viewed as a recap of partial definitions given and points being made in the other contribution.

5 - LAYERING OF FUNCTIONS

The use of structured layering, as advocated in several contributions (see section 4) makes it clear that all contributions are rather consistent :

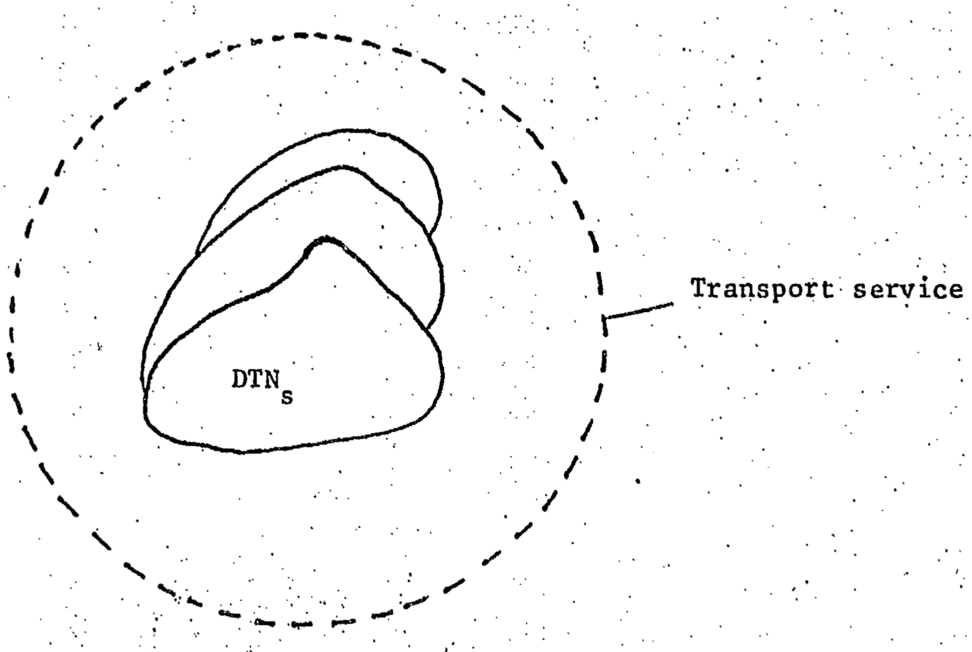
a) A Data Transmission Network (DTN) provides communication between end systems (DTEs). A variety of DTN's is recognized, each one with its specific interface (e.g. circuit switched DTNs with X 21 or packet switched DTNs with X 25, or private DTNs with their own interface).



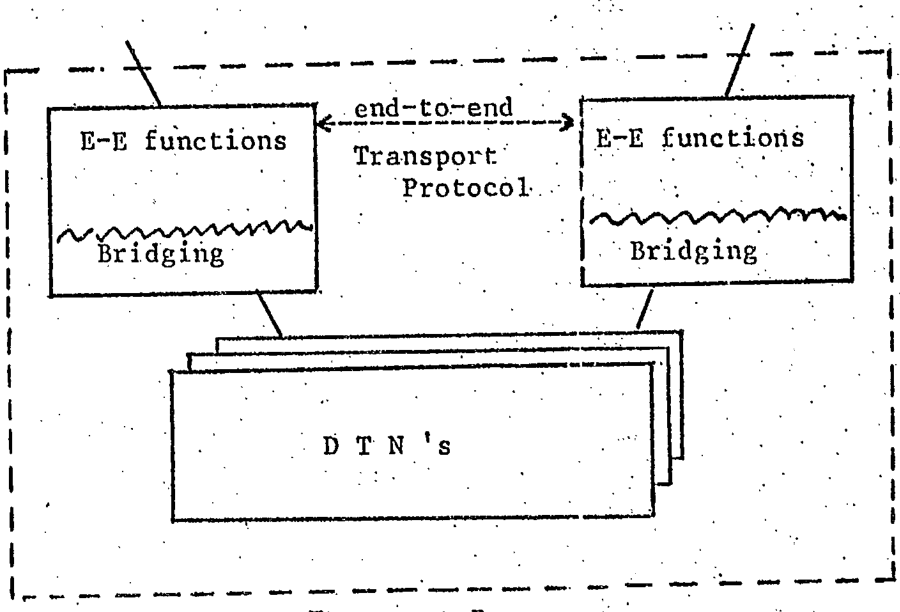
It should be noted that, with respect to that structure, HDLC is being viewed either as an end-to-end protocol (on a circuit interconnecting two (or more) DTE's), or as part of an interface protocol (between a DTE and a DCE using X 25).

That basic component of information networks architecture is identified as such in most contributions (N 1132, 1167, 1272, 1277, 1280, 1283, 1296, 1309, 1310, 1350, 1369). The internal structure of DTNs is not analysed, but it is recognized that different types of DTNs have to be considered. Moreover, it is implicitely admitted in all contributions (related to that aspect of Project 17) that the data transmission service is symmetric, i.e. both ends view the same service since they are at the same level of the architecture.

b) The next level in the architecture, on which a general consensus appears (N 1167, 1173, 1272, 1277, 1280, 1281, 1282, 1283, 1296, 1309, 1350, 1369) is frequently referred to as the Transport level. According to the definition proposed in N 1369, it should be termed "Transport Service". The corresponding black box is described as providing processes with the possibility of exchanging data with each other. It is explicitely stated in several contributions (N 1167, 1272, 1280, 1283, 1296, 1369) and implicitely admitted in others, that the transport service should be the same, whatever data transmission network is being used, to keep higher layers independent from variations in data transmission media. Elements of definition of such a service are proposed in N 1368.



c) Between the DTN service and the Transport service is located the transport end-to-end protocol according to (N 1272, 1280, 1281, 1282, 1283, 1350, 1368). Other contributions (N 1132, 1259, 1277, 1286, 1306) describe elementary functions that need to be performed to provide the transport service, some of them being end to end functions, some others being interface functions. Tentative lists of such functions can be found in N 1309 and 1368. It is clear that a bridging layer is necessary to absorb variations in services and interfaces provided by DTN's (e.g. logical channel ID or interface fragmentation when using X 25). It is also clear that some functions are clearly end to end and DTN independent (e.g. process addressing, or end-to-end error/ flow control).



Transport Box

d) On top of the transport layer, a set of Function oriented Protocols are identified, among which terminal access is the most frequently listed (N 1167, 1272, 1277, 1280, 1283, 1350, 1369). It is obvious that this has some relationship with the work on PAD undertaken recently within CCITT and that coordination in that area between ISO and CCITT is urgently needed.
Other function oriented protocols include file transfer, remote job entry and functions related to operating systems.

6 - POINTS FOR STUDY

The following list, derived from contributions on Project 17, Communication Systems Architecture and End-to-End Protocols, summarizes points of agreement and areas of study :

Point 1 : Architecture

Should define an overal architecture in terms of levels and services. There is already a general agreement on the existence of a data transmission level, a transport level and function oriented levels.
A precise definition of services is needed and the architecture must be detailed, namely in function oriented layers.

Point 2 : Bridging

Interface functions and DTN dependent functions need to be clearly identified. The latter will likely need to be carefully studied to provide a common ground base for end-to-end protocols.

Point 3 : End-to-End Transport

End-to-end functions and DTN independent functions need to be clearly identified and standardized. They will provide transport services to function oriented protocols.

Point 4 : Terminal access

Terminal access protocols are likely to be the most urgent function oriented protocols to be standardized within ISO.

access to both P/S & C/S services

Need more specific identification

Point 5 : Other function oriented protocols

Coordination is required with other SCs within TC 97, namely SC 2 "Character sets and coding", SC 5 "Programming language", SC 14 "Representation of Data Elements" and SC 15 "Labelling and file structure".

Point 6 :

- Packet Net
- PADS

Standardization of network architecture and end-to-end protocols within ISO also requires coordination with CCITT, since this body is working on equivalent problems (e.g. terminal access) and users should not be faced with different standards depending on which supplier provides the equipment or the service.

APPENDIX A

LIST OF ISO/TC 97/SC 6 REFERENCE DOCUMENTS

Document Number-Suffix	Title	Date assigned
771	United Kingdom discussion paper on systems architecture.	
798	Report of working Group on "Communication Systems Architecture"	June 1973
921	The 'systems architecture' technique for the description of complex data communications systems	August 1974
1016	Resolutions of the TOKYO meeting	October 1974
1132	HDLC Message/Packet heading formats	July 1975
1167	United Kingdom contribution on higher level protocols	
1172	HDLC, Information field structure	September 1975
1173	Working document from ECMA TC 9 to ISO/TC 97/ SC 6	September 1975
1248	HDLC, Information field labelling	October 1975
1249	Minutes of meeting of AD HOC WG 1	October 1975
1258	Resolutions	October 1975
1259	Australian contribution on standardization of information field format (resolution 13,	

1272	Contributions on End-to-End Protocols	March 1976
1277	ECMA Working Paper on a Possible Information Field Structure for Function Oriented and User Levels.	April 1976
1280	IFIP Contribution on Projects 15, 17 and 24 (Pocket Switching)	April 1976
1281	IFIP Contribution on Projects 15, 17 and 24 Inter Network Communication	April 1976
1282	IFIP Contribution on Projects 17, 24 (End to End Protocol)	April 1976
1283	French contribution on Project n° 17	May 1976
1286	HDLC Heading Formats, ISO/TC 97/SC 6, Project 24	April 1976
1296	Japanese comments on Information Field Format and Function and End-to-End Protocol	May 1976
1308	USA contribution on Headings	October 1976
1309	Rapporteurs' Report on Project 24	October 1975
1310	USA contribution to Project 17, End-to-End Protocol	October 1976
1331	Draft agenda of the SYDNEY SC 6 meeting	November 1976
1338	Australian contribution to project 24 - Standardization of information field format and function for communication purposes	November 1976
1342	Minutes of the meeting of ISO/TC 97/SC 6/WG 1	November 1976

1350	Higher level or end-to-end protocols (project 17, 24)	December 1976
1356	Report of the preparatory ISO/TC 97/SC 6/ working group 2 meeting, Utrecht	January 1977
1358	Meeting notice and draft agenda (SC 6/WG 2)	March 1977
1367	National activity report	January 1977
1368	Definition of the transport layer in the end-to-end protocols architecture (french contribution on end-to-end protocols - project 17	January 1977
1369	Basic architecture for end-to-end protocols (french contribution on end-to-end protocols- project 17)	January 1977

ISO TC/97/SC 15

	User-user protocols for interchange of data over communication networks	March 1976
1400	USA	
1408	France	