EUROPEAN TELECOMMUNICATION STANDARD

ETS 300 079

October 1991

Source: ETSI TC-TE

Reference: DE/TE-01010

ICS: 33.080

Key words: ISDN; Videotex; End-to-end protocols; Circuit mode DTE - DTE

Integrated Services Digital Network (ISDN); Syntax-based Videotex end-to-end protocols Circuit mode DTE - DTE

ETSI

European Telecommunications Standards Institute

ETSI Secretariat

Postal address: F-06921 Sophia Antipolis CEDEX - FRANCE
Office address: 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE
X.400: c=fr, a=atlas, p=etsi, s=secretariat - **Internet:** secretariat@etsi.fr

Tel.: +33 92 94 42 00 - Fax: +33 93 65 47 16

New presentation - see History box

Whilst every care has been taken in the preparation and publication of this document, errors in content, typographical or otherwise, may occur. If you have comments concerning its accuracy, please write to "ETSI Editing and Committee Support Dept." at the address shown on the title page.

Contents

Foreword

This European Telecommunication Standard (ETS) has been produced by the Terminal Equipment (TE) Technical Committee of the European Telecommunications Standards Institute (ETSI).

This ETS aims to meet the urgent requirements of network operators and equipment manufacturers who are designing equipment to operate on an Integrated Services Digital Network (ISDN).

This ETS also has a close relationship to ETSs 300 072 [10] to 300 076 [14] inclusive and to other proposed ETSs currently under development within the TE Technical Committee (draft prETS 300 177 [19], draft prETS 300 149 [18] and final draft prETS 300 080 [15]). Full details of these documents are given in Clause 2 (Normative references) of this document.

NOTE: Abstract testing requirements for this ETS are to be developed by ETSI.

Blank page

1 Scope

This standard specifies the end-to-end protocols for the ISDN syntax-based Videotex Service.

NOTE 1: This standard does not specify any service aspects of the ISDN syntax-based Videotex Service.

This standard is applicable to devices supporting the ISDN syntax-based Videotex, to be attached at either side of a T reference point or coincident S and T reference points when used as an access to the public ISDN. In this context, a device is either an ISDN Videotex Terminal, an ISDN Videotex Service Centre, an ISDN Videotex Access Point or an ISDN Videotex Host (cf. subclause 3.1).

For the lower layer protocols (layer 1 to layer 3), this standard makes use of final draft prETS 300 080 [15] (based on CCITT Recommendation T.90) for demand circuit-switched calls using the 64 kbit/s unrestricted digital information bearer capability and the DTE/DTE case of the Network Layer peer entities in B-channel connection.

NOTE 2: The end-to-end protocol specified in this ETS is intended to be applicable not only for the DTE/DTE connection over an ISDN.

2 Normative references

This ETS incorporates by dated or undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references subsequent amendments to, or revisions of, any of these publications apply to this ETS only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies.

[1] CCITT Recommendation F.300 (1988): "Videotex Service".

[2] CCITT Recommendation T.51 (1988): "Coded character sets for telematic services".

[3] CCITT Recommendation X.3: "Packet assembly/disassembly facility (PAD) in a public data network".

[4] CCITT Recommendation X.25 (1988): "Interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) for terminals operating in the packet mode and connected to public data networks by dedicated circuit".

[5] CCITT Recommendation X.29: "Procedures for the exchange of control information and user data between a packet assembly/disassembly (PAD) facility and a packet mode DTE or another PAD".

[6] CCITT Recommendation X.75 (1984): "Packet-switched signalling system between public networks providing data transmission services".

[7] CCITT Recommendation X.121 (1988): "International numbering plan for public data networks".

[8] Draft prETS 300 011: "Integrated Services Digital Network (ISDN); Primary rate user-network interface, Layer 1 specification and test principles (T/L 03-14)".

[9] Draft prETS 300 012: "Integrated Services Digital Network (ISDN); Basic user-network interface, Layer 1 specification and test principles (T/L 03-07)".

[10] ETS 300 072 (1990): "Terminal Equipment (TE); Videotex Presentation Layer protocol, Videotex presentation layer data syntax (T/TE 06-01)".

[11] ETS 300 073 (1990): "Terminal Equipment (TE); Videotex presentation layer syntax; Geometric Display (CEPT Rec. T/TE 06-02, Edinburgh 1988)".

[12] ETS 300 074 (1990): "Terminal Equipment (TE); Videotex Transparent Data (CEPT Rec. T/TE 06-03, Edinburgh 1988)".

[13] ETS 300 075 (1990): "Terminal Equipment (TE); Videotex processable data (T/TE 06-04)".

[14] ETS 300 076 (1990): "Terminal Equipment (TE); Videotex, Terminal Facility Identifier (TFI) (T/TE 06-05)".

[15] Final draft prETS 300 080: "Integrated Services Digital Network (ISDN); ISDN lower layer protocols for telematic terminals (T/TE 12-04)".

[16] ETS 300 102-1 (1990): "Integrated Services Digital Network (ISDN); User-network interface layer 3, Specifications for basic call control".

[17] ETS 300 125 (1991): "Integrated Services Digital Network (ISDN); User-network interface data link layer specifications, Application of CCITT Recommendations Q.920/I.440 and Q.921/I.441".

[18] Draft prETS 300 149: "Terminal Equipment (TE); Videotex: Audio Syntax".

[19] Draft prETS 300 177: "Terminal Equipment (TE); Videotex, Photographic Syntax (T/TE 06-06)".

[20] ISO 7776: "Information processing systems - Data communications - High-level data link control procedures - Description of the X.25 LAPB-compatible DTE data link procedures".

[21] ISO 8208: "Information processing systems - Data communications - X.25 Packet Level Protocol for Data Terminal equipment".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of this standard, the following definitions apply:

Access Function: the functional entity which gives access to the Videotex Service. This entity is an integral part of the Videotex Service.

Access Network: the network which provides the link between the Terminal Function and the Access Function.

Audio data: generic term for data which can be made audible (e.g. wave form encoded, phonemically encoded).

Distributed Videotex Application: a Videotex Application which makes use of more than one Videotex Host.

Host Access Network: the network which provides the link between the Access Function and the Host Function(s). It is an integral part of the Videotex Service and may be void.

Host Function: the abstraction of the Videotex Applications available in a particular Videotex Service.

Multi-media communication: term used to indicate that both pictorial and audio data are exchanged.

Pictorial data: generic term for data which can be displayed (e.g. alphamosaic, geometric, photographic).

Telematic Command: In the scope of this ETS, a Telematic Command is a specific service element which is carried in a complete packet sequence of X.25 PLP DATA packets with the Q-bit set to 1. The first octet in the User Data field of the first packet of the complete packet sequence carries a value which is reserved for videotex (cf. draft CCITT Recommendation X.29 (1992), "Telematic service message, videotex").

Terminal Function: the abstraction of a functional entity which acts as a Videotex Terminal.

Videotex Access Point: see CCITT Recommendation F.300 [1].

Videotex Application: see CCITT Recommendation F.300 [1].

Videotex External Host: see CCITT Recommendation F.300 [1].

Videotex Host: this term describes a computer which offers one or more applications and/or facilities. It can be represented through a Videotex Host Computer, an External Videotex Host or a Videotex Service Centre.

Videotex Host Computer: see CCITT Recommendation F.300 [1].

Videotex Service: see CCITT Recommendation F.300 [1].

Videotex Service Centre: see CCITT Recommendation F.300 [1].

NOTE: According to CCITT Recommendation F.300, a Videotex Service Centre provides host and/or access functions, i.e. it may also act as a Videotex Access Point.

Videotex Terminal: see CCITT Recommendation F.300 [1].

3.2 Abbreviations

For the purposes of this standard, the following abbreviations apply:

AU	**A**ccess **U**nit
CCITT	**I**nternational **T**elegraph and **T**elephone **C**onsultative **C**ommittee
CD	**C**all **D**eflection
CLIP	**C**alling **L**ine **I**dentification **P**resentation
CLIR	**C**alling **L**ine **I**dentification **R**estriction
DCE	**D**ata **C**ircuit-terminating **E**quipment
DDI	**D**irect **D**ialling **I**n
DDU	**D**ialogue **D**ata **U**nit
DFK	**D**efinable **F**unction **K**ey
DTE	**D**ata **T**erminal **E**quipment
ETS	**E**uropean **T**elecommunication **S**tandard
HAN	**H**ost **A**ccess **N**etwork
IB	**I**n-**B**and
ISDN	**I**ntegrated **S**ervices **D**igital **N**etwork
ISO	**I**nternational **O**rganization for **S**tandardization
ISPBX	**I**ntegrated **S**ervices **P**rivate **B**ranch e**X**change
IVAP	**I**SDN syntax-based **VAP**
IVT	**I**SDN syntax-based **VT**
LSB	**L**east **S**ignificant **B**it
MSB	**M**ost **S**ignificant **B**it
MSN	**M**ultiple **S**ubscriber **N**umber
OB	**O**ut-**B**and
PDN	**P**ublic **D**ata **N**etwork
PDU	**P**rotocol **D**ata **U**nit
PH	**P**acket **H**andler
PLP	**P**acket **L**evel **P**rotocol
PSPDN	**P**acket **S**witched **P**ublic **D**ata **N**etwork
PSTN	**P**ublic **S**witched **T**elephone **N**etwork
Rec	**Rec**ommendation
SBV	**S**yntax-**B**ased **V**ideotex
SUB	**Sub**addressing
TDU	**T**elesoftware **D**ata **U**nit
TFI	**T**erminal **F**acility **I**dentifier
TPD	**T**ransparent **P**rocessable **D**ata
UUI	**U**ser-to-**U**ser **I**nformation
UUS	**U**ser-to-**U**ser **S**ignalling
VAP	**V**ideotex **A**ccess **P**oint
VC	**V**irtual **C**ircuit
VH	**V**ideotex **H**ost
VPDE	**V**ideotex **P**resentation **D**ata **E**lement
VS	**V**ideotex **S**ervice
VT	**V**ideotex **T**erminal

4 Overview

The main feature of an ISDN syntax-based Videotex Terminal (VT) is the capability to access those Videotex Services which are defined for and used in a Public Switched Telephone Network (PSTN) or Packet Switched Public Data Network (PSPDN) environment using an ISDN as an access network.

More advanced services can make use of additional features (specific for an ISDN), which are presented in the following subclauses.

4.1 Direct selection of Videotex applications

The protocol allows the Videotex Terminal to shortcut the dialogue and welcome phase of the Videotex Service and to be connected with a given Videotex Application identified by a network address or a mnemonic.

4.2 Multi-communication

Multi-communication allows a terminal to establish several independent communication channels to one or several independent Videotex Hosts or Videotex Applications.

The protocol supports multi-communication using virtual circuits on one B-channel.

NOTE: The handling of several independent communication channels may appear in any configuration (cf. informative Annex A).

EXAMPLE: Assume that "ETSI" and "CCITT" are two Videotex Applications offering the possibility to Videotex users to read texts of telecommunication standards. In order to compare the text of a given ETS with an equivalent recommendation, the terminal should establish a virtual circuit (VC) with each of the two applications. The information from the two applications shall be simultaneously displayed on the terminal using some windowing capability.

4.3 Multi-media communication

Multi-media information may contain pictorial data (i.e. alphamosaic, geometric, and photographic display elements) and audio data. The audio data may be transmitted either in parallel or in serial with the pictorial data.

When the audio data is transmitted in serial with the pictorial data, a single virtual circuit may be used to carry both.

When the audio data has to be sent in parallel with the pictorial data, a second virtual circuit dedicated to the transmission of audio data shall be set up.

The protocol supports multi-media communication using virtual circuits on one B-channel, providing for the simultaneous reproduction of both pictorial and audio information, in addition to a serial transmission already covered by the data syntax.

NOTE: The handling of several additional communication channels may appear in any configuration (cf. informative Annex A).

4.4 Distributed Videotex Application

It is recognized that, to accomplish some specific application, the application may be decomposed into several sub-applications which may be implemented on more than one host. A communication channel from the terminal to each of the sub-applications may exist. The incoming datastreams are combined in the terminal to deliver the desired service to the user.

The protocol supports distributed videotex applications using virtual circuits on one B-channel.

EXAMPLE: Assume an application which serves PSTN terminals and ISDN terminals at the same time. The PSTN application uses only alphamosaic information, while the ISDN application additionally offers photographic pictures. The "alphamosaic-host" is used for the dialogue with both PSTN and ISDN terminals. When a specific picture has to be displayed, on request of the "dialogue-host" a second VC is established between the ISDN terminal and the "picture-host".

4.5 ISDN supplementary services

The protocol allows for a use of several ISDN supplementary services (Multiple Subscriber Number (MSN), Subaddressing (SUB), User-to-User Signalling 1 (UUS 1), Call Deflection (CD), Calling Line Identification Presentation (CLIP) and Calling Line Identification Restriction (CLIR)), mainly to support ISDN-specific Videotex Service selection mechanisms. For details on these supplementary services, refer to Annex B (informative).

4.6 Handling of incoming calls

The protocol permits an ISDN syntax-based Videotex Terminal to be called by another terminal or a Videotex Service.

In the case of a called ISDN syntax-based Videotex Terminal, the handling of ISDN supplementary services is not within the scope of this ETS.

5 Configurations

Various configurations and topologies may be used, examples of which are given in Annex A (informative). It shall be the responsibility of the Videotex Service providers to opt for the appropriate configuration(s) in the definition of the ISDN syntax-based Videotex Service.

6 General model

In order to describe all configurations, the following reference models are used:

- a communication model;
- a terminal model;
- an access network scenario.

The communication model describes the relationships between the functional entities involved in the communication. It models the communication between a user and a Videotex Service in terms of terminal functions, access network functions and service functions.

The terminal model defines, in an abstract way, the terminal aspects which are relevant for the protocol.

The access network scenario describes the use of an ISDN as the Access Network.

6.1 Communication model

In describing the communication model, the following entities can be identified:

- a User;
- a Terminal Function;
- an Access Network;
- an Access Function;
- a Host Access Network;
- a Host Function.

In all videotex configurations, the **User** employs a terminal to communicate with a Videotex Service. The abstraction of the terminal is named **Terminal Function**.

The **Access Network** provides the link between the Terminal Function and the Videotex Service. It may consist of one or several networks of different types. The Access Network shall be an ISDN within the scope of this ETS.

NOTE 1: For an overview of possible Access Network architectures for the DTE/DCE case of connection over an ISDN refer to DE/TE-01012.

The **Access Function** is the functional entity which gives access to the Videotex systems. It is an integral part of the Videotex Service.

The **Host Access Network**, which is an optional functional entity, is the abstraction of the function, which connects the Access Function to one or more Host Functions. Depending on the actual topology used in a specific service, it may be void.

The **Host Function** is the abstraction of the collection of Videotex Applications.

The interfaces and protocols defined between the functional entities are:

(1) User interface;
(2) Terminal-Access Network interface;
(3) Access Function-Access Network interface;
(4) Access Function-Host Access Network interface;
(5) Host Function-Host Access Network interface;
(6) Terminal-Access Function protocol;
(7) Host Access protocol.

Figure 1 shows the relationships between the entities:

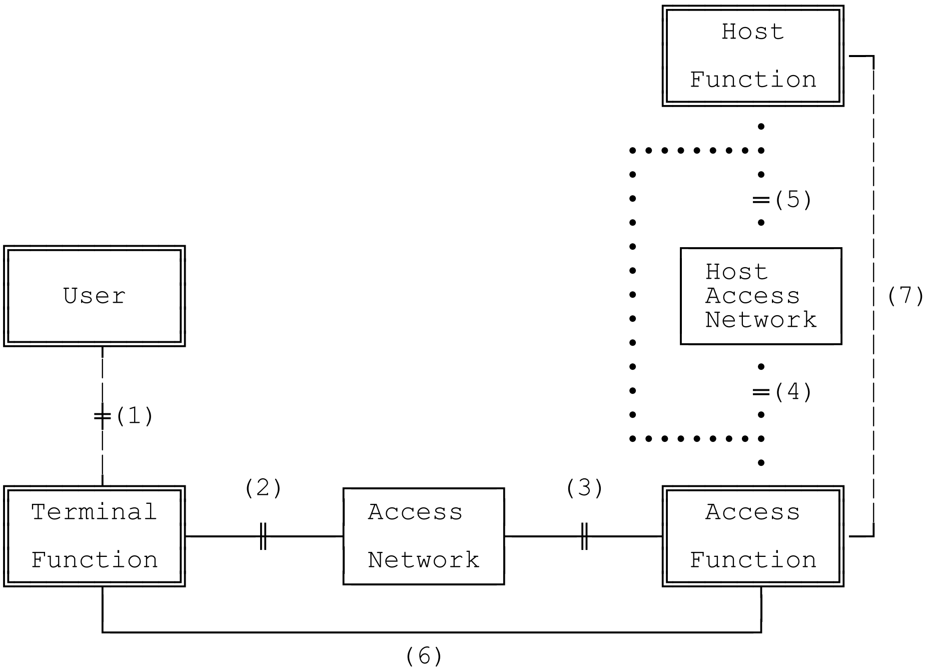


Figure 1: Communication Model

Of these interfaces and protocols, only items (2), (3) and (6) are within the scope of this ETS.

The Terminal-Access Network interface (2) and the Access Function-Access Network interface (3) are covered by the layer 1 to layer 3 protocol specified in this ETS. For the DTE/DTE case of connection, it shall be based on draft prETS 300 080 [15].

The Terminal-Access Function protocol (6), the end-to-end protocol between Terminal and Access Function, specifies the features of a Videotex Terminal which can be used for the ISDN syntax-based Videotex Service.

NOTE 2: The same end-to-end protocol may be used between two terminals, whereas one of the terminals acts as an Access Function (cf. subclause 7.1 and informative Annex A, clause A.7).

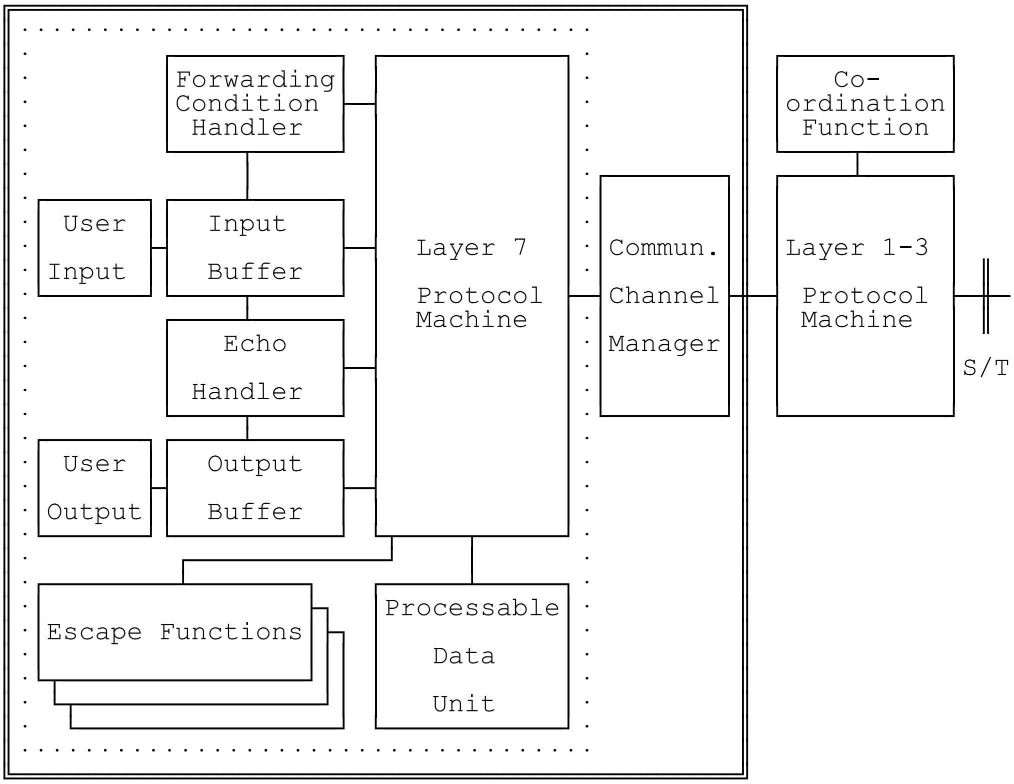
6.2 Terminal model

The terminal model is the reference model which is used throughout the document. This model describes the relationships between functions of an ISDN syntax-based Videotex Terminal.

The Application layer is decomposed into the following components:

- communication channel manager;
- layer 7 protocol machine;
- input buffer;
- output buffer;
- echo handler;
- forwarding condition handler;
- processable data unit;
- escape function units.

Figure 2 introduces these functions and their relationship:



NOTE: For every communication channel, one separate independent instance of each of the functional entities inside the dotted rectangle exists.

Figure 2: Functional entities of the terminal model

The protocol machine handling layers 1-3 of the protocol and the co-ordination function are integral parts of the Network layer entity and do not require further refinement in the model.

NOTE 1: The co-ordination function may also be empty.

In the context of this ETS, a **communication channel** is equivalent to a virtual circuit (VC) on an established B-channel.

NOTE 2: For other cases of connections, a communication channel might be defined differently.

The **communication channel manager** supervises the handling of related communication channels. The protocol allows for the definition of a hierarchical relationship between communication channels. An application can request the establishment of further dependent communication channels via an already established communication channel. The requesting communication channel shall be named a "Master"-channel and the requested communication channels shall be named "Slave"-channels.

NOTE 3: The handling of more than one communication channel is optional. Therefore, the communication channel manager may also be empty.

This relationship is maintained during the lifetime of a communication channel implying that existing "Slave"-channels have to be closed implicitly when the "Master"-channel is closed. The communication channel manager takes care of this implicit closure and keeps track of related communication channels.

For each established communication channel, one separate independent (virtual) instance of the following functional entities exists (regardless of whether it is an independent or a linked communication channel).

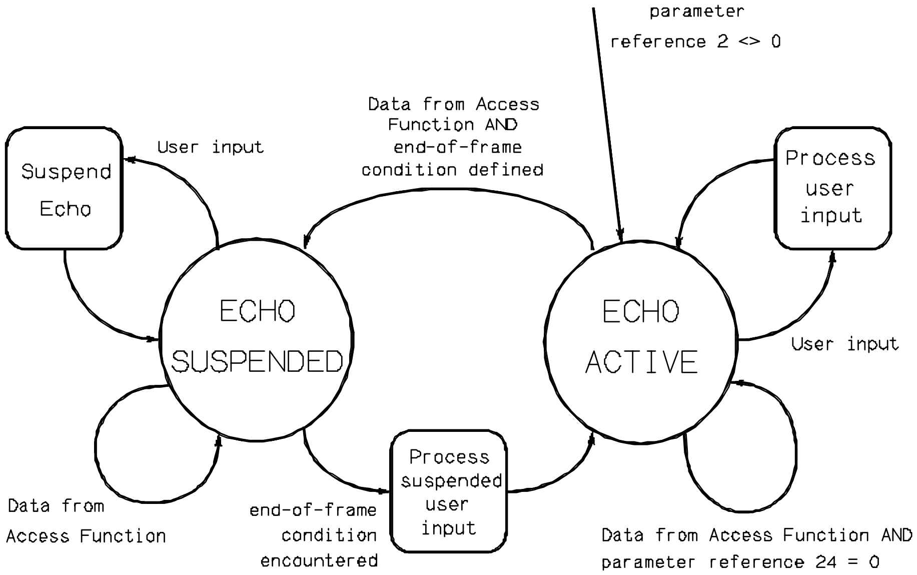
The **layer 7 protocol machine** supports the procedures as defined in the Application layer protocol. It conveys information to the output buffer (display data), the echo handler (parameter settings, related to echoing), the forwarding condition handler (parameter settings, related to forwarding and function key information), the communication channel manager (related communication channels) or the processable data or escape modules.

The **input buffer** receives the user input, which subsequently is forwarded to the protocol machine in order to be transmitted to the Access Function, and which may be used in parallel by the echo handler. A terminal may locally edit this buffer. The origin of data entered in the input buffer need not be defined, having no relevance for the description of the protocol of this ETS.

NOTE 4: The specification of local editing is outside the scope of this ETS.

The **output buffer** receives display data from both the layer 7 protocol machine and the echo handler. The processing of display data is not described in the model having no relevance for the description of the protocol of this ETS.

The **echo handler** controls the echoplex procedures. With regard to echoing, the state diagram in the following figure 3 shall apply:



NOTE: An end-of-frame condition is defined if either CCITT Recommendation X.3 [3] extended parameter reference 24 is not implemented or if parameter reference 24 has been set to a non-zero value.

Figure 3: Echo handler state diagram

When echoing is enabled, the initial terminal state shall be ECHO_ACTIVE. The initial state of echoing is defined in Annex C (normative).

In the state ECHO_ACTIVE, all user input shall be echoed immediately. Echoing shall be performed on a character basis in the sense of ETS 300 072 [10], subclause 2.1.

As long as no data have been received from the Access Function, the terminal remains in the state ECHO_ACTIVE.

When the terminal is in the state ECHO_ACTIVE and data are received from the Access Function, one of the following shall apply:

- if an end-of-frame condition has been set (CCITT Recommendation X.3 [3] extended parameter reference 24 is not implemented or set to a non-zero value, cf. clause 10) the terminal shall enter the state ECHO_SUSPENDED;

- if no end-of-frame condition has been set (parameter reference 24 is set to 0), the terminal shall remain in the state ECHO_ACTIVE.

When the terminal is in the state ECHO_SUSPENDED, data received from the Access Function shall be processed (e.g., displayed) and the echoing of user input shall be suspended until an end-of-frame has been detected.

When an end-of-frame has been detected in the ECHO_SUSPENDED state, suspended user input shall be processed (e.g., displayed) immediately prior to any other action and the terminal shall return to the ECHO_ACTIVE state.

The possible end-of-frame conditions are:

- the receipt of a complete packet sequence shall be an end-of-frame condition, if CCITT Recommendation X.3 [3] extended parameter reference 24 is not implemented;

- end-of-frame conditions as defined by parameter reference 24.

  Certain possibly defined end-of-frame conditions can be checked for by inspecting the contents of the output buffer;

- the receipt of SBV_service elements different from the SBV_VTX_Data element are considered to be implicit end-of-frame conditions.

The **forwarding condition handler** manages the forwarding conditions and upon user input forwards data to the layer 7 protocol machine. It also takes care of the currently defined user input field length (cf. subclause 10.1.2.1).

NOTE 5: The handling of the input field is a local matter.

The forwarding conditions can be divided into the following groups in a hierarchical order (i.e., lower numbered conditions have precedence):

1) X.29 PAD messages (mandatory);
2) X.3 idle timer expiration (mandatory);
3) X.3 standard forwarding conditions (mandatory);
4) X.3 extended forwarding conditions (optional);
5) Definable forwarding conditions (optional).

NOTE 6: It is a local matter to decide at which point in time the Application layer data are assembled into X.25 packets. Nevertheless, it is expected that these packets are forwarded whenever a packet is full.

CCITT Recommendation X.29 [5], §2.1, specifies the receipt of a PAD message as a forwarding condition. PAD messages are used to set and read parameters.

If the idle timer is set to values > 0, the contents of the input buffer shall be forwarded if the input buffer has remained unchanged for the period of time as defined by the current value of the idle timer parameter.

CCITT Recommendation X.3 [3] standard forwarding conditions define single octets which trigger forwarding when encountered in the input buffer. The default X.3 forwarding conditions are given in Annex C (normative).

CCITT Recommendation X.3 [3] extended forwarding conditions (cf. clause 10) define fixed multi-octet sequences, which are in use in videotex systems, accessed over the PSTN or the PSPDN (cf. Annex E (informative)).

Definable function keys can be set by the service to define arbitrary forwarding conditions. The support of definable function keys is optional. These conditions are additional to the CCITT Recommendation X.3 [3] forwarding conditions. In cases of an ambiguity between these different sets of forwarding conditions, the CCITT Recommendation X.3 [3] conditions shall have the higher priority.

A terminal supporting definable function keys shall be capable of handling a minimum of 8 function keys with identifications 1 to 8. These function keys may be linked to forwarding conditions. An optional user visible name element may be used to distinguish the functions for the user. The function key definitions may be changed by sending a new definition for some specific key, or deleted by either overwriting them with an empty definition (zero length code sequence) or by resetting all definable function keys.

EXAMPLE: To adapt an ISDN syntax based videotex terminal to a system using function keys INDEX, HELP, DELETE, CORRECT, PREVIOUS, NEXT, REPEAT and SEND in a Videotex Service X, forwarding conditions may be downloaded with key 1 being INDEX, key 2 being HELP etc. and corresponding codes (e.g., 1/3 x/y). To use this terminal in Videotex Service Y using the same function keys, but other code sequences (e.g., the code sequence for CORRECT shall be 0/8 instead of 1/3 4/7) forwarding conditions are downloaded according to the environment.

A terminal supporting definable function keys may abstain from treating a character sequence as a forwarding condition, if it is entered directly into the input buffer, even though this character sequence is identical to an already defined forwarding condition code sequence linked to a function key. By definition, this rule applies only to definable forwarding conditions as these provide a way to link forwarding conditions to function keys. CCITT Recommendation X.3 [3] forwarding conditions are not subject to this rule. The default state of the list of function keys is specified in Annex C (normative).

NOTE 7: This rule eases implementations, because otherwise terminals should be capable of interpreting long sequences of input characters.

The **processable data unit** handles processable data as defined in ETS 300 075 [13]. It may also be capable of handling processable data transparently without making use of the Dialogue Data Units (DDUs). The default regime to be used for conveying processable data is defined in Annex C (normative).

The **escape functions** handle services which are not specified in this ETS. The default states of processes related to SBV_Escape subservices are defined in Annex C (normative).

6.3 Access network scenario

Figure 4 below illustrates the access network scenario for the DTE/DTE case of connection. The ISDN provides for an end-to-end circuit-switched link between the Terminal Function and the Access Function.

NOTE: For an overview of access network scenarios for the DTE/DCE case of connection over an ISDN refer to DE/TE-01012.

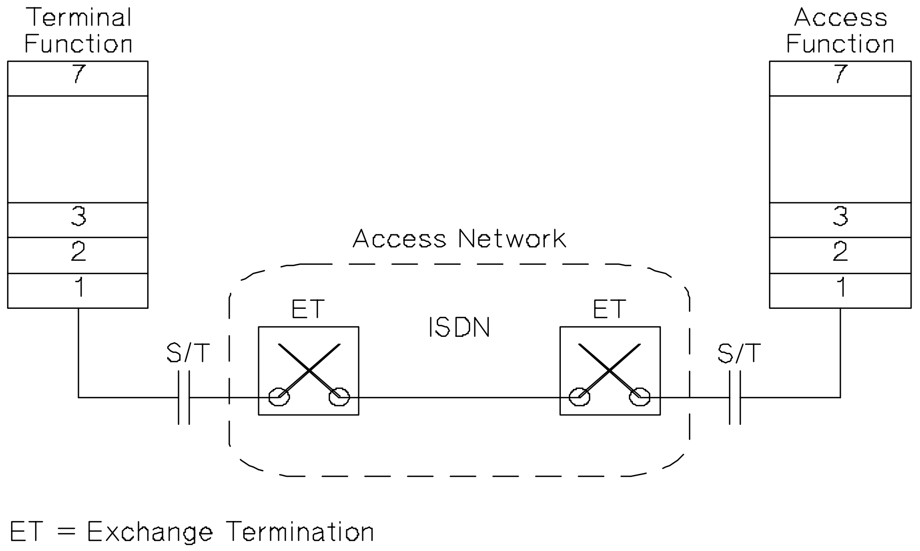


Figure 4: Access Network scenario

6.4 Protocol pillar

7	ETS 300 079 (this ETS) ETS 300 072 to 300 076 ([10] to [14]) ETS 300 149 [18] ETS 300 177 [19]	
6	null (NOTE 1)	
5	null	
4	null	
3	ETS 300 102-1 [16] (Q.931)	ISO 8208 [21]
2	ETS 300 125 [17] (Q.921)	X.75 [6] (NOTE 2) or ISO 7776 [20]
1	ETS 300 011 [8] (I.431) or ETS 300 012 [9] (I.430)	
Layer	D-channel	B-channel

NOTE 1: The main purpose of layer 6, the conversion from the "abstract syntax" to the "transfer syntax" is not necessary, because in this case the abstract data syntax in layer 7 is identical to the transfer data syntax. Also all other features of the layer 6 are not used and therefore "null" is inserted for layer 6. The abstract syntax in layer 7 and the coding in layer 6 correspond to the data syntaxes DS I, DS II and DS III in CCITT Recommendation T.101.

NOTE 2: CCITT Recommendation X.75 [6] as modified in final draft prETS 300 080 [15].

Figure 5: Protocol Pillar

The protocol pillar used for the DTE/DTE case of connection is given in figure 5.

NOTE: The protocol pillars for the DTE/DCE case of connection over an ISDN are given in DE/TE-01012.

The lower layer (layer 1 to layer 3) protocols are as specified in final draft prETS 300 080 [15] for the DTE/DTE connection using ISDN circuit mode. This ETS gives additional application rules in Clause 11.

The Application layer protocol is defined in this ETS. The data syntax used for ISDN syntax-based Videotex is defined in ETSs 300 072 to 300 076 ([10] to [14]), in draft prETS 300 149 [18], and in draft prETS 300 177 [19].

7 Service definition

7.1 Introduction

The ISDN syntax-based Videotex Application layer incorporates some functionalities related to those Videotex Services originally defined for the PSTN/PSPDN environment, as well as functionalities related to the additional features provided by an ISDN.

This Clause describes, in an abstract way, the services offered by the ISDN syntax-based Videotex Application Layer in terms of:

- the primitive actions and events of the service;
- the parameter data associated with each primitive action and event;
- the relationship between, and the valid sequence of these actions and events.

The services are divided into two classes:

- Mandatory Kernel services (Table 1);
- Optional services (Table 2).

Table 1: Kernel services (mandatory)

Kernel Service	C/NC	Initiated by TF/AF	Function
SBV_Establish	C	TF	Connection establishment
SBV_Release	NC	Both	Connection release
SBV_Reset	C	AF	Reset to basic state
SBV_VTX_Data	NC	Both	Videotex data transfer
SBV_Set_Param	NC	AF	Set X.3 parameters
SBV_Read_Param	NC	AF	Read X.3 parameters
SBV_Set_Read_Param	NC	AF	Set and read X.3 parameters
SBV_Param_Indication	NC	TF	X.3 parameter indication
SBV_TFI	C	AF (NOTE)	Exchange of Terminal Facility Identifier (TFI) information
SBV_TC_Error	NC	Both	Telematic Command Error Message

Abbreviations:

C	Confirmed service
NC	Non-confirmed service
AF	Access Function
TF	Terminal Function

NOTE: The support of the SBV_TFI service element shall be mandatory for ISDN syntax-based Videotex Terminals only. It shall be up to the Videotex Service to employ this terminal capability.

Table 2: Optional Services

Telematic Command	C/NC	Initiated by TF/AF	Function
SBV_Channel_Open (NOTE 1)	C	AF	Invitation to establish an additional communication channel
SBV_Channel_Close (NOTE 1)	C	AF	Invitation to release an additional communication channel
SBV_Channel_Error (NOTE 1)	NC	TF	Additional communication channel error message
SBV_Begin_Application	C	TF	Direct selection of a Videotex Application
SBV_End_Application	NC	TF	Return to the Videotex Application selection phase
SBV_TPD_Begin (NOTE 2)	C	Both	Begin of transparent processable data exchange regime
SBV_TPD_End (NOTE 2)	NC	Both	End of transparent processable data exchange regime
SBV_DFK	NC	AF	Definition of function keys
SBV_Escape	NC	Both	Services not standardized in this ETS

Abbreviations:

C	Confirmed service
NC	Non-confirmed service
AF	Access Function
TF	Terminal Function

NOTE 1: If implemented, all three communication channel management services shall be supported.
NOTE 2: If implemented, both transparent processable data (TPD) services shall be supported.

For every service described below, parameters may be mandatory or optional. Mandatory parameters shall always be present within the primitives used to convey the service element and shall be coded as defined in Clause 9 of this ETS. Optional parameters need not be present within the primitive used to convey the service element; if an optional parameter is present, it shall be coded according to Clause 9 of this ETS; if it is not present, no information shall be associated to this event (i.e., no default value is defined for any parameter).

The protocol is used between a Terminal Function and an Access Function and as such is not symmetrical, because some services shall be invoked by the Terminal Function or the Access Function only.

The default role assignment shall be defined as follows: the **calling** side (i.e., the entity which issues an SBV_Establish Request) shall act as the Terminal Function; the **called** side (i.e., the entity which receives an SBV_Establish Indication) shall act as the Access Function.

NOTE: This role assignment may be reversed upon indication by the caller, but only if the called side accepts (cf. subclause 7.2.1).

7.2 Kernel services

7.2.1 SBV_Establish

The SBV_Establish service element shall be used to establish a communication. It may be initiated by the Terminal Function only. This service is confirmed.

An ISDN syntax-based Videotex Terminal shall not be obliged to handle an incoming call.

NOTE 1: This service allows for a direct selection of a Videotex Application. This direct selection may also be accomplished using the SBV_Begin_Application service, once the communication establishment has been successful.

NOTE 2: This service may be used to establish several virtual circuits. It is left to the terminal to decide whether to open an additional B-channel or to establish another virtual circuit on the existing B-channel.

NOTE 3: A Videotex Service calling a SBV Terminal acts, in terms of the communication model (cf. subclause 6.1), as a Terminal Function, at least up to that point in time when both sides agreed to reverse the role assignment.

The primitives and parameters of the SBV_Establish service are described in table 3 below.

Table 3

Parameters	Request	Indication	Response	Confirm
OB_Called_Address	Optional	Optional(=)	--	--
OB_Called_Subaddress	Optional	Optional(=)	--	--
OB_Application_Address	Optional	Optional(=)	--	--
OB_Application_Selection	Optional	Optional(=)	--	--
OB_Application_Data	Optional	Optional(=)	--	--
OB_User_Data	Optional	Optional(=)	--	--
IB_Called_Address	Optional	Optional(=)	--	--
IB_Application_Address	Optional	Optional(=)	--	--
IB_Application_Selection	Optional	Optional(=)	--	--
IB_Application_Data	Optional	Optional(=)	--	--
IB_User_Data	Optional	Optional(=)	--	--
Reference	Optional	Optional(=)	--	--
Reverse_Role_Assignment	Optional	Optional(=)		
Result	--	--	Optional	Optional(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request or Response primitive

Parameters:

OB_Called_Address: this parameter carries the ISDN address of the Access Function to be reached; it shall be present when the first connection is to be established.

OB_Called_Subaddress: this parameter carries the ISDN subaddress of the Access Function; it may only be present if the OB_Called_Address parameter is present.

OB_Application_Address: this parameter carries complementary information about the Videotex Application to be reached; this may include an ISDN address or a network address which is interpreted by the Access Function.

OB_Application_Selection: this parameter carries complementary information (e.g. mnemonic) about the Videotex Application to be reached; this is interpreted by the Access Function.

OB_Application_Data: this parameter carries data to be sent to the Videotex Application to be reached; it is passed transparently to the Host Function via the Access Function.

OB_User_Data: this parameter carries user data not specified in this ETS which is interpreted by the Access Function.

IB_Called_Address: this parameter carries a Network address; it represents the address of the Videotex Application to be reached and is interpreted by the Access Function.

IB_Application_Address: this parameter carries complementary information about the Videotex Application to be reached; this may include an ISDN address or a network address; it is interpreted by the Access Function.

IB_Application_Selection: this parameter carries complementary information (e.g. mnemonic) about the Videotex Application to be reached; it is interpreted by the Access Function.

IB_Application_Data: this parameter carries data to be sent to the Videotex Application to be reached; it is passed transparently to the Host Function via the Access Function.

IB_User_Data: this parameter carries user data not specified in this ETS which is interpreted by the Access Function.

Reference: this parameter carries a reference value as given by a previously received SBV_Channel_Open Indication. This parameter shall not be present when no reference value is available (cf. subclause 7.3.1).

Reverse_Role_Assignment: this parameter has no value. When present, it shall act as an indicator that the calling side requests to reverse the default role assignment (i.e., the calling side requests to act as the Access Function instead of the Terminal Function). The inverted role assignment applies upon a "Successful" confirmation. A called side which is not able to support a reversed role assignment may choose to ignore or reject the incoming call.

Result: this parameter carries the result of the communication establishment; it may take the value "Successful" or, in the case of an attempt to establish a communication with a Videotex Application directly via the Access Function, it may take the value "Application not available" if the communication to the Access Function has been successfully established, but the indicated Videotex Application could not be reached.

NOTE: This parameter may only be present if the CCITT Recommendation X.25 [4] Fast Select facility is used (cf. subclause 11.3).

7.2.2 SBV_Release

The SBV_Release service shall be used either to release a communication or to indicate that a previous attempt to establish a connection using the SBV_Establish service has not been successful. It may be initiated by both sides at any time. Moreover, in case of an Access Network failure, both sides shall receive an SBV_Release indication. In the case of the release of a "Master"-channel, all the associated "Slave"-channels shall be released (cf. subclause 6.2 of this ETS).

This service is non-confirmed. The primitives and parameters of the SBV_Release service are described in table 4 below.

Table 4

Parameters	Request	Indication
OB_Cause	Optional	Optional(=)
IB_Cause	Optional	Optional(=)
IB_Diagnostic	Optional	Optional(=)
Result	Optional	Optional(=)
NOTE: Either OB_Cause or IB_Cause/IB_Diagnostic shall be present in an Indication.		

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

OB_Cause: this parameter carries information about the reason for the release of the communication on the D-channel.

IB_Cause: this parameter carries the reason for the release of the communication on the B-channel.

IB_Diagnostic: this parameter carries additional information on the reason for the release of the communication on the B-channel.

Result: this parameter carries the application dependent reason for the release of the communication; it may take the values "Application not available" or "Application already engaged" as the result of a previous attempt to establish a communication with a Videotex Application directly via the Access Function.

NOTE: This parameter can only appear if the CCITT Recommendation X.25 [4] Fast Select facility is used.

7.2.3 SBV_Reset

The SBV_Reset service element shall be used by the Access Function to reset the Terminal Function into the basic state as defined in Annex C (normative). In addition, all possibly existent "Slave"-channels (cf. subclause 6.2) related to the communication channel on which the service request was received shall be released. This service is confirmed.

Parameters: none.

7.2.4 SBV_VTX_Data

This service element shall be used to transfer videotex data. It may be initiated by both sides. This service is non-confirmed. The primitives and parameters of the SBV_VTX_Data service are described in the table 5.

Table 5

Parameters	Request	Indication
VTX_Data	Mandatory	Mandatory(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

VTX_Data: this parameter carries the Videotex data to be transmitted.

7.2.5 X.3 [3] parameter manipulation services

TEMPORARY NOTE: The term "CCITT Recommendation X.3 [3] parameters" as used in this clause refers to **all** X.3 parameters as defined in Clause 10 (cf. table 46, TEMPORARY NOTE 1).

7.2.5.1 SBV_Set_Param

This service element shall be used to set one or more of the CCITT Recommendation X.3 [3] parameters of the Terminal Function. It may be initiated by the Access Function only. This service is non-confirmed. The primitives and parameters of the SBV_Set_Param service are described in table 6 below.

Table 6

Parameters	Request	Indication
X.3_Parameter_List	Optional	Optional(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

X.3_Parameter_List: this parameter carries the X.3 [3] parameter(s) to be modified.

7.2.5.2 SBV_Read_Param

This service element shall be used to read one or more of the CCITT Recommendation X.3 [3] parameters of the Terminal Function. It may be initiated by the Access function only. This service is non-confirmed. The primitives and parameters of the SBV_Read_Param service are described in table 7 below.

Table 7

Parameters	Request	Indication
X.3_Parameter_List	Optional	Optional(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

X.3_Parameter_List: this parameter carries the X.3 parameter(s) to be read.

7.2.5.3 SBV_Set_Read_Param

This service element shall be used to set one or more of the CCITT Recommendation X.3 [3] parameters of the Terminal Function and to read the indicated parameters. It may be initiated by the Access function only. This service is non-confirmed. The primitives and parameters of the SBV_Set_Read_Param service are described in table 8 below.

Table 8

Parameters	Request	Indication
X.3_Parameter_List	Optional	Optional(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

X.3_Parameter_List: this parameter carries the X.3 [3] parameter(s) according to CCITT Recommendation X.29 [5].

7.2.5.4 SBV_Param_Ind

This service element shall be used to indicate the CCITT Recommendation X.3 [3] parameters values of the Terminal Function. It may be initiated by the Terminal Function only. This service is non-confirmed. The primitives and parameters of the SBV_Param_Ind service are described in table 9 below.

Table 9

Parameters	Request	Indication
X.3_Parameter_List	Optional	Optional(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

X.3_Parameter_List: this parameter carries the CCITT Recommendation X.3 [3] parameter(s) to be modified.

7.2.6 SBV_TFI

The SBV_TFI service shall be used by the Access Function only to request for the facilities of the Terminal Function. This service is confirmed. The primitives and parameters of the SBV_TFI service are described in table 10 below.

Table 10

Parameters	Request	Indication	Response	Confirm
TFI_Enq	Mandatory	Mandatory(=)	--	--
TFI_Ack	--	--	Mandatory	Mandatory(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request or Response primitive

Parameters:

TFI_Enq: this parameter carries the TFI request data.

TFI_Ack: this parameter carries the TFI response data.

7.2.7 SBV_TC_Error

The SBV_TC_Error service shall be used by the Application layer itself to indicate to the application that the previously received Telematic Command was erroneous (syntactical error or SBV_Service not supported or inopportune incoming PDU). This service is non-confirmed. The primitives and parameters of the SBV_TC_Error service are described in table 1 below.

NOTE: As defined in subclause 8.2.7, a TC_Error_PDU is never an inopportune incoming PDU, in the sense of this protocol. It therefore does never cause another TC_Error_PDU, but acts as a (negative) response PDU.

Table 11

Parameters	Request	Indication
Error_Detection_Location	Mandatory	Mandatory(=)
Error_Code	Mandatory	Mandatory(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

Error_Detection_Location: this parameter contains the information whether the error has been detected by the local or the remote Application layer.

Error_Code: this parameter carries information about the reason of the error in the Telematic Command or indicates that the previously received Telematic Command is not supported.

7.3 Optional services

7.3.1 Communication channel management services

The following service elements allows an Access Function to request the Terminal Function to establish or release additional communication channels (i.e., virtual circuits). If implemented, all communication channel management services shall be supported.

The additional communication channels may be **independent** from or **linked** with the communication channels which carried the request to open them. The distinction between the two types of additional communication channels is possible by using the Reference parameter of the SBV_Channel_Open service:

Independent additional communication channel: No Reference parameter shall be present in the SBV_Channel_Open Request.

Linked additional communication channel: a Reference parameter shall be present in the SBV_Channel_Open service. In that case, the requesting communication channel shall be named a "Master"-channel and the requested communication channel (when opened) shall be named a "Slave"-channel.

The reference value carried in the Reference parameter shall be associated to one and only one opened "Slave"-channel. When the "Slave"-channel has been established by the Terminal Function, the reference value used to establish this communication channel shall be no longer available.

7.3.1.1 SBV_Channel_Open

This service element shall be used by the Access Function to invite the Terminal Function to open an additional communication channel. The opening of the additional communication channel is accomplished by using the SBV_Establish service. The SBV_Channel_Open service is confirmed. The primitives and parameters of the SBV_Channel_Open service are described in table 12 below.

Table 12

Parameters	Request	Indication	Response	Confirm
Req_OB_Called_Address	Optional	Optional(=)	--	--
Req_OB_Called_Subaddress	Optional	Optional(=)	--	--
OB_Application_Address	Optional	Optional(=)	--	--
OB_Application_Selection	Optional	Optional(=)	--	--
OB_Application_Data	Optional	Optional(=)	--	--
Req_OB_User_Data	Optional	Optional(=)	--	--
Ind_OB_Cause	--	--	Optional	Optional(=)
Req_IB_Called_Address	Optional	Optional(=)	--	--
IB_Application_Address	Optional	Optional(=)	--	--
IB_Application_Selection	Optional	Optional(=)	--	--
IB_Application_Data	Optional	Optional(=)	--	--
Req_IB_User_Data	Optional	Optional(=)	--	--
Packet_Size	Optional	Optional(=)	--	--
Window_Size	Optional	Optional(=)	--	--
Ind_IB_Cause	--	--	Optional	Optional(=)
Ind_IB_Diagnostic	--	--	Optional	Optional(=)
Reference	Optional	Optional(=)	--	--
Result	--	--	Mandatory	Mandatory(=)

NOTE 1: At least one of the following parameters shall be present in the Request:

- Req_OB_Called_Address;
- OB_Application_Address;
- OB_Application_Selection;
- Req_OB_User_Data
- Req_IB_Called_Address;
- IB_Application_Address;
- IB_Application_Selection
- Req_IB_User_Data.

NOTE 2: The introduction of a parameter to indicate the request to establish a separate B-channel is for further study.

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request or Response primitive.

Parameters:

Req_OB_Called_Address: carries the ISDN address of the Access Function to be reached; it can be mapped onto the OB_Called_Address parameter of the SBV_Establish service which shall be used to open the additional communication channel.

Req_OB_Called_Subaddress: carries the ISDN subaddress of the Access Function to be reached; it can be mapped onto the OB_Called_Subaddress parameter of the SBV_Establish service.

OB_Application_Address: carries complementary information (e.g. ISDN address or network address) about the Videotex Application to be reached; it can be mapped onto the OB_Application_Address parameter of the SBV_Establish service.

OB_Application_Selection: carries complementary information (e.g. mnemonic) about the Videotex Application to be reached; it can be mapped onto the OB_Application_Selection parameter of the SBV_Establish.

OB_Application_Data: carries data to be passed to the Videotex Application to be reached; it can be mapped onto the OB_Application_Data parameter of the SBV_Establish.

Req_OB_User_Data: carries data to be passed transparently to the requested Application; it can be mapped onto the OB_User_Data parameter of the SBV_Establish service.

Ind_OB_Cause: carries the reason for the unsuccessful additional communication channel establishment on the D-channel; it corresponds to the OB_Cause parameter of the SBV_Release service.

Req_IB_Called_Address: this parameter carries the Network address of the Videotex Application to be reached; it can be mapped onto the IB_Called_Address parameter of the SBV_Establish service which shall be used to open the additional communication channel.

IB_Application_Address: this parameter carries complementary information (e.g. ISDN address or network address) about the Videotex Application to be reached; it can be mapped onto the IB_Application_Address parameter of the SBV_Establish service.

IB_Application_Selection: carries complementary information (e.g. mnemonic) about the Videotex Application to be reached; it can be mapped onto the IB_Application_Selection parameter of the SBV_Establish service.

IB_Application_Data: carries data to be passed to the Videotex Application to be reached; it can be mapped onto the IB_Application_Data parameter of the SBV_Establish service.

Req_IB_User_Data: carries data to be passed transparently to the requested Application; it can be mapped onto the IB_User_Data parameter of the SBV_Establish service.

Packet_Size: carries a requested layer 3 packet size for the establishment of the additional communication channel. If the requested value is not available for any reason, the value which is currently in use on the requesting channel may be employed instead.

Window_Size: carries a requested layer 3 window size for the establishment of the additional communication channel. If the requested value is not available for any reason, the value which is currently in use on the requesting channel may be employed instead.

Ind_IB_Cause: carries the reason for the unsuccessful additional communication channel establishment on the B-channel; it corresponds to the IB_Cause parameter of the SBV_Release service.

Ind_IB_Diagnostic: carries additional information about the reason for the unsuccessful additional communication channel establishment on the B-channel; it corresponds to the IB_Diagnostic parameter of the SBV_Release service.

Reference: carries a reference information to be sent transparently to the Videotex Application to be reached; this parameter can be mapped onto the Reference parameter of the SBV_Establish service.

Result: carries the result of the opening of the additional communication channel; it may take the values "Successful", "Not successful" or "Application not available".

If the Result parameter value is "Successful", no Ind_OB_Cause, Ind_IB_Cause or Ind_IB_Diagnostic parameters shall be present.

7.3.1.2 SBV_Channel_Close

The SBV_Channel_Close service element shall be used by the Access Function only to request the release of a previously opened additional "Slave"-channel. Only the request of a channel to release one of its associated "Slave"-channels shall be considered as valid; the release is accomplished by using the SBV_Release service. All other requests shall result in a negative confirmation but shall be otherwise ignored.

This SBV_Channel_Close service is confirmed. The primitives and parameters of the SBV_Channel_Close service are described in table 13 below.

Table 13

Parameters	Request	Indication	Response	Confirm
Reference	Mandatory	Mandatory(=)	--	--
Result	--	--	Mandatory	Mandatory(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request or Response primitive

Parameters:

Reference: this parameter carries the reference of the communication channel to be closed; it takes the value of the Reference parameter of the SBV_Channel_Open service element previously used to request the establishment of the linked additional communication channel.

Result: this parameter carries the result of the release; it may take the value "Successful" if the Release succeeded or "Illegal Reference parameter" if the Reference parameter was erroneous; it may also take the value "Release not allowed" in case of a "Slave"-channel trying to request the closure of its "Master"-channel.

7.3.1.3 SBV_Channel_Error

The SBV_Channel_Error service shall be used by the Terminal Function only to inform the Access Function via the "Master"-channel that a previously opened "Slave"-channel has been disconnected for some reason. This service is non-confirmed. The primitives and parameters of the SBV_Channel_Error service are described in table 14 below.

Table 14

Parameters	Request	Indication
Reference	Mandatory	Mandatory(=)
Ind_OB_Cause	Optional	Optional(=)
Ind_IB_Cause	Optional	Optional(=)
Ind_IB_Diagnostic	Optional	Optional(=)
NOTE: Either Ind_OB_Cause or Ind_IB_Cause/ Ind_IB_Diagnostic shall be present in an Indication.		

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

Reference: this parameter carries the reference of the disconnected communication channel; it takes the value of the Reference parameter of the SBV_Channel_Open service element previously used to open the linked additional communication channel.

Ind_OB_Cause: this parameter carries the reason for the disconnection on the D-channel of the referenced communication channel.

Ind_IB_Cause: this parameter carries the reason for the disconnection on the B-channel of the referenced communication channel.

Ind_IB_Diagnostic: this parameter carries additional information on the reason for the disconnection on the B-channel of the referenced communication channel.

7.3.2 Application selection services

7.3.2.1 SBV_Begin_Application

The SBV_Begin_Application service shall be used by the Terminal Function only to select a Videotex Application directly by short-cutting the dialogue and welcome phase in the Access Function. If a connection to another Videotex Application existed prior to the use of this service, then this connection may be closed (depending on the Videotex Service definition) by the Access Function before trying to establish the requested connection. This service is confirmed.

NOTE: This service does not preclude a direct application selection using the SBV_Establish service.

The primitives and parameters of the SBV_Begin_Application service are described in table 15.

Table 15

Parameters	Request	Indication	Response	Confirm
IB_Application_Address	Optional	Optional(=)	--	--
IB_Application_Selection	Optional	Optional(=)	--	--
IB_Application_Data	Optional	Optional(=)		
IB_User_Data	Optional	Optional(=)	--	--
Result	--	--	Mandatory	Mandatory(=)
NOTE: At least one of the following parameters shall be present in the Request: - IB_Application_Address; - IB_Application_Selection; - IB_User_Data.				

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request or Response primitive

Parameters:

IB_Application_Address: this parameter carries the address of the Videotex Application to be reached; it is interpreted by the Access Function in order to establish the communication with the requested Application.

IB_Application_Selection: this parameter carries information (e.g. mnemonic) about the Videotex Application to be reached; it is interpreted by the Access Function in order to establish the communication with the requested Application.

IB_Application_Data: this parameter carries data to be passed transparently by the Access Function to the Videotex Application to be reached .

IB_User_Data: this parameter carries data not specified in this ETS to be interpreted by the Access Function.

Result: this parameter carries the result of the direct access to the Application; it may take the value "Successful", "Application not available" if the requested Videotex Application is not available or "Application already engaged" for those Access Functions which are not able to disconnect from an engaged application upon reception of an SBV_Begin_Application Indication.

7.3.2.2 SBV_End_Application

The SBV_End_Application service shall be used by the Terminal Function only to disconnect from a Host Function and to return to a dialogue and welcome phase. This service is non-confirmed.

Parameters: none

7.3.3 Transparent processable data services

The SBV_TPD (Transparent Processable Data) services may be used to allow for the exchange of processable data (cf. ETS 300 075 [13]) transparently without making use of the Dialogue Data Units (DDUs). If implemented, both transparent processable data services shall be supported.

7.3.3.1 SBV_TPD_Begin

The SBV_TPD_Begin service shall be used to request for the exchange of transparent processable data. It may be initiated by both sides. This service is confirmed. The primitives and parameters of the SBV_TPD_Begin service are described in table 16 below.

Table 16

Parameters	Request	Indication	Response	Confirm
Result	--	--	Mandatory	Mandatory(=)
DDU_Fall-Back_Mode	--	--	Optional	Optional(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request or Response primitive

Parameters:

Result: this parameter carries the information, whether the receiving side shall support the transparent transmission of processable data or not. It may take the values "Successful" or "TPD not supported".

If the Result parameter takes the value "Successful", then the processable data shall be exchanged transparently and the following application rules shall apply:

- the DDU Layer shall not be used;
- a T-Associate request shall be issued immediately upon the receipt of an SBV_TPD_Begin Confirmation indicating "Successful";
- a TDU is carried in one single complete packet sequence of DATA packets with the Q-bit set to 0;
- a complete packet sequence may contain one and only one TDU;
- 8-bit processable data are exchanged via DATA packets with the Q-bit set to 0;
- the regime of an SBV_TPD_Begin lasts up to the transmission of an SBV_TPD_End Request.

If the Result parameter takes the value "TPD not supported", then the processable data shall be exchanged as defined in ETS 300 075 [13] using the SBV_VTX_Data service, if possible. The regime of transparent processable data exchange shall be implicitly terminated.

DDU_Fall-Back_Mode: This parameter may be present if the value of the Result parameter indicates "TPD not supported". It may give additional information about supported DDU modes on the receiving side. This parameter may take values from "A" to "G" and their permitted combinations (cf. subclause 9.4.5.2), corresponding to the respective DDU modes as specified in ETS 300 075 [13].

7.3.3.2 SBV_TPD_End

The SBV_TPD_End service shall be used to end the regime of a previously established transparent processable data exchange. This service is non-confirmed.

The receipt of an SBV_TPD_End Indication shall release any possibly pending T-Associate regime in the sense of ETS 300 075 [13].

Parameters: none

7.3.4 SBV_DFK

The SBV_DFK (Definable Function Keys) service shall be used to define function keys. It shall be initiated by the Access Function only. This service is non-confirmed. The primitives and parameters of the SBV_DFK service are described in table 17 below.

Table 17

Parameters	Request	Indication
Function_Keys	Optional	Optional(=)
Reset_Keys	Optional	Optional(=)

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

At least one of the parameters should be used. If no parameter is present in an SBV_DFK Request, the service primitive shall be ignored.

Reset_Keys: this parameter indicates that all definable function keys set by a previous SBV_DFK invocation are to be cleared. Forwarding conditions set by CCITT Recommendation X.3 [3] set commands are not influenced by this reset mechanism.

Function_Keys: this parameter carries a list of function key definitions. Each function key definition consists of an identification, a user-visible name, an indication whether pressing the function key shall be a forwarding condition, and a code sequence (a sequence of octets which has to be sent to the Access Function when the forwarding condition occurs).

7.3.5 SBV_Escape

The SBV_Escape service is used for functions which exist in some European Videotex Services (e.g., bulk update, transport of binary data). For details concerning this service refer to Annex D (informative). This service is non-confirmed. The primitives and parameters of the SBV_Escape service are described in table 18 below.

Table 18

Parameters	Request	Indication
Service-ID	Mandatory	Mandatory(=)
Data	Optional	Optional

(=): the value of this parameter is identical to the one of the corresponding parameter in the preceding Request primitive

Parameters:

Service-ID: this parameter carries the identification of a specific service (cf. subclause 9.2 and Annex D (informative)).

Data: this parameter carries data not standardised in this ETS.

8 Protocol

8.1 Introduction

The protocol is based on CCITT Recommendation X.29 [5] and uses the ISO 8208/X.25 Packet Level Protocol as defined in final draft prETS 300 080 [15]. It allows for the use of standard PAD messages as described in this ETS.

The service primitives are associated with PDUs on the Network Layer which are conveyed via the D-channel, or via X.25 packets as described in ISO 8208 [21] on an established B-channel.

NOTE 1: For other types of connections, the associated PDUs on the Network Layer may be conveyed differently.

Consequently, the procedures to be used are those defined in ISO 8208 [21] with application rules defined in subclause 11.3 of this ETS unless otherwise specified. Moreover, when CCITT Recommendation X.25 [4] packets carry CCITT Recommendation X.29 [5] PAD messages, the procedures described in CCITT Recommendation X.29 [5] shall apply.

CCITT Recommendation X.29 [5] related protocol errors (e.g., unrecognizable X.29 messages) shall be treated in accordance to CCITT Recommendation X.29 [5]. Telematic Command related errors shall be treated by the protocol as defined in this ETS.

Except for the SBV_Establish service, all service primitives shall be issued only on an established communication channel.

On a given communication channel, all confirmed services shall proceed in accordance to the following rules:

- after having issued a Request-PDU, the sending side Application layer shall not issue any other PDU (except for the SBV_Release-Request-PDU or the SBV_Reset-Request-PDU or the TC_Error-PDU) before the reception of the Response-PDU;

- upon reception of a Request-PDU, the receiving side Application layer shall not issue any other PDU (except the for the SBV_Release-Request-PDU or the SBV_Reset-Request-PDU) than the Response-PDU to the requested service;

- after the reception of a Request-PDU, any reception of another Request-PDU (except for the SBV_Release-Request-PDU or the SBV_Reset-Request-PDU) before the issuing of the Response shall be treated as a SBV-protocol error;

- a SBV protocol error detected by the Application layer will cause the Application layer to send TC_Error-PDU to the remote Application layer. This TC_Error-PDU replaces the outstanding Response-PDU, which shall not be issued by the Application layer, even if the Response primitive already occurred at the upper service interface;

- after having issued a Request-PDU, the reception of either a SBV_Release-Request-PDU or a SBV_Reset-Request-PDU or a TC_Error-PDU implicitly results in a (negative) confirmation;

- at the issuing time of confirmed service Request-PDU, a timer shall be started; when the timer runs out and the Response-PDU has not been received, the communication shall be released.

  The initial value of this timer shall not be smaller than timer value T28 multiplied by counter value R28 as defined in ISO 8208 [21].

  NOTE 2: It is a local matter, and therefore out of the scope of the protocol definition, how the given requirements are fulfilled locally, especially with respect to the upper service interface, its primitives and its user application.

8.2 Kernel services

8.2.1 SBV_Establish

The protocol elements providing the SBV_Establish service are carried by D-channel messages and/or layer 3 PDUs as defined in tables 19 and 20 below, depending on the communication state and the parameters in the request.

Table 19: D-channel messages

Service Primitive	D-channel message	§ in ETS 300 102-1 [16]
SBV_Establish Request	SETUP	3.1.16
SBV_Establish Indication	SETUP	3.1.6
SBV_Establish Response	CONNECT	3.1.4
SBV_Establish Confirm	CONNECT	3.1.4

Table 20: Layer 3 PDUs

Service Primitive	Protocol Data Unit	§ in ISO 8208 [21]
SBV_Establish Request	CALL REQUEST	12.2.1
SBV_Establish Indication	INCOMING CALL	12.2.1
SBV_Establish Response	CALL ACCEPTED	12.2.2
SBV_Establish Confirm	CALL CONNECTED	12.2.3

Procedures:

The detailed procedures to set up a communication are described in final draft prETS 300 080 [15]; additional application rules are specified in subclause 11.1 of this ETS.

8.2.2 SBV_Release

The protocol elements providing the SBV_Release service are carried by D-channel messages and/or layer 3 PDUs as defined in tables 21 and 22 below, depending on the communication state and the parameters in the request.

Table 21: D-channel messages

Service Primitive	D-channel message	§ in ETS 300 102-1 [16]
SBV_Release Request	DISCONNECT	3.1.6
SBV_Release Indication	DISCONNECT or RELEASE or RELEASE COMPLETE	3.1.6 3.1.11 3.1.12

Table 22: Layer 3 PDUs

Service Primitive	Protocol Data Unit	§ in ISO 8208 [21]
SBV_Release Request	CLEAR REQUEST	12.2.3
SBV_Release Indication	CLEAR INDICATION	12.2.3

Procedures:

The detailed procedures to release a communication are described in final draft prETS 300 080 [15]; additional application rules are specified in subclause 11.1 of this ETS.

8.2.3 SBV_Reset

The protocol elements providing the SBV_Reset service are carried by CCITT Recommendation X.25 [4] PLP PDUs of ISO 8208 [21] as described in table 23 below.

Table 23

Service	Protocol Data Unit	§ in ISO 8208 [21]
SBV_Reset Request	RESET REQUEST	12.5
SBV_Reset Indication	RESET INDICATION (NOTE 1)	12.5
SBV_Reset Response	RESET CONFIRMATION	12.5
SBV_Reset Confirmation (NOTE 2)	RESET CONFIRMATION	12.5

NOTE 1: The reaction of an access function on receipt of a RESET INDICATION is outside the scope of this ETS.

NOTE 2: The issuing of this primitive may or may not be related to the completion of the X.25/PLP Reset procedure. Any data or qualified data following the SBV_Reset Confirmation is transmitted after completion of the X.25/PLP Reset procedure.

Procedures:

The detailed procedures to reset a communication are described in draft prETS 300 080 [15]; additional application rules are specified in subclause 11.3 of this ETS.

8.2.4 SBV_VTX_Data

The protocol elements providing the SBV_VTX_Data service are carried by CCITT Recommendation X.25 [4] PLP PDUs of ISO 8208 [21] as described in table 24 below.

Table 24

Service Primitive	Protocol Data Unit	§ in ISO 8208 [21]
SBV_VTX_Data Request	DATA	12.3.1
SBV_VTX_Data Indication	DATA	12.3.1

Procedures:

VTX_Data are conveyed in a complete packet sequence. The detailed procedure to exchange VTX_Data is described in ISO 8208 [21], Clause 6; additional application rules are specified in final draft prETS 300 080 [15] and in subclause 11.1 of this ETS.

8.2.5 X.3 [3] parameter manipulation services

TEMPORARY NOTE: The term "CCITT Recommendation X.3 [3] parameters" as used in this clause refers to **all** X.3 parameters as defined in clause 10 (cf. table 46, TEMPORARY NOTE 1).

8.2.5.1 SBV_Set_Param

The protocol elements providing the SBV_Set_Param service are carried by CCITT Recommendation X.29 [5] PDUs as described in table 25 below.

Table 25

Service Primitive	Protocol Data Unit	§ in CCITT X.29 [5]
SBV_Set_Param Request	DATA (Q=1)	4.4.5
SBV_Set_Param Indication	DATA (Q=1)	4.4.5

Procedures:

The detailed procedure to set one or more of the available CCITT Recommendation X.3 [3] parameters of the Terminal Function is described in the CCITT Recommendation X.29 [5], §§1.5, 3.1 and 3.5.1.

8.2.5.2 SBV_Read_Param

The protocol elements providing the SBV_Read_Param service are carried by CCITT Recommendation X.29 [5] PDUs as described in table 26 below.

Table 26

Service Primitive	Protocol Data Unit	§ in CCITT X.29 [5]
SBV_Read_Param Request	DATA (Q=1)	4.4.5
SBV_Read_Param Indication	DATA (Q=1)	4.4.5

Procedures:

The detailed procedure to read one or more of the available CCITT Recommendation X.3 [3] parameters of the Terminal Function is described in the CCITT Recommendation X.29 [5], §§1.5, 3.1 and 3.5.1.

8.2.5.3 SBV_Set_Read_Param

The protocol elements providing the SBV_Set_Read_Param service are carried by CCITT Recommendation X.29 [5] PDUs as described in table 27 below.

Table 27

Service Primitive	Protocol Data Unit	§ in CCITT X.29 [5]
SBV_Set_Read_Param Request	DATA (Q=1)	4.4.5
SBV_Set_Read_Param Indication	DATA (Q=1)	4.4.5

Procedures:

The detailed procedure to set and read one or more of the available CCITT Recommendation X.3 [3] parameters of the Terminal Function is described in the CCITT Recommendation X.29 [5], §§1.5, 3.1 and 3.5.1.

8.2.5.4 SBV_Param_Ind

The Terminal Function shall issue one and only one SBV_Param_Ind Request, when it is required due to a previously received SBV_Set_Param Indication or SBV_Read_Param Indication or a SBV_Set_Read_Param Indication. The protocol elements providing the SBV_Set_Read_Param service are carried by CCITT Recommendation X.29 [5] PDUs as described in table 28 below.

Table 28

Service Primitive	Protocol Data Unit	§ in CCITT X.29 [5]
SBV_Param_Ind Request	DATA (Q=1)	4.4.5
SBV_Param_Ind Indication	DATA (Q=1)	4.4.5

Procedures:

The detailed procedure to indicate one or more of the available CCITT Recommendation X.3 [3] parameters of the Terminal Function is described in the CCITT Recommendation X.29 [5], §§1.5, 3.1 and 3.5.1.

8.2.6 SBV_TFI

The protocol elements providing the SBV_TFI service are carried by a Telematic Command as described in table 29 below.

Table 29

Service	Telematic Command	Reference
SBV_TFI	SBV_TFI_TC	9.2 in this ETS

Procedures:

Sending: a SBV_TFI Request primitive results in an SBV_TFI_TC PDU; the Access Function is waiting for an SBV_TFI Confirm.

Receiving: a valid incoming SBV_TFI_TC PDU results in an SBV_TFI Indication primitive; the Terminal Function issues a SBV_TFI Response which results in sending an SBV_TFI_TC PDU.

8.2.7 SBV_TC_Error

The protocol elements providing the SBV_TC_Error service are carried by a Telematic Command as described in table 30 below.

Table 30

Service	Telematic Command	Reference
SBV_TC_Error	SBV_TC_Error_TC	9.2 in this ETS

Procedures:

Sending: A SBV_TC_Error_TC PDU is sent by the Application layer when a previously received Telematic Command PDU was either not supported or erroneous. This does not apply for any previously received TC_Error PDU (regardless whether its parameters have been valid or not). At the same time, a TC_Error_Indication is given to the local application with the Error_Detection_Location parameter set to "local".

Receiving: a valid incoming SBV_TC_Error_TC PDU results in an SBV_TC_Error Indication primitive with the Error_Detection_Location parameter set to "remote".

8.3 Optional services

The protocol elements providing for the optional services are carried by Telematic Commands.

8.3.1 Communication channel management services

8.3.1.1 SBV_Channel_Open

The protocol elements providing the SBV_Channel_Open service are carried by a Telematic Command as described in table 31 below.

Table 31

Service	Telematic Command	Reference
SBV_Channel_Open	SBV_Channel_Open_TC	9.2 in this ETS

Procedures:

Sending: an SBV_Channel_Open primitive results in sending an SBV_Channel_Open_TC PDU; after which the Access Function is waiting for an SBV_Channel_Open Confirm positive or negative.

Receiving: a valid incoming SBV_Channel_Open_TC PDU results in an SBV_Channel_Open Indication. The Terminal Function tries to establish the requested communication using the SBV_Establish service; when the establishment of the requested communication has been terminated (successfully or not) the Terminal Function issues a SBV_Channel_Open Response primitive which results in sending an SBV_Channel_Open_TC PDU with the appropriate result code.

8.3.1.2 SBV_Channel_Close

The protocol elements providing the SBV_Channel_Close service are carried by a Telematic Command as described in table 32 below.

Table 32

Service	Telematic Command	Reference
SBV_Channel_Close	SBV_Channel_Close_TC	9.2 in this ETS

Procedures:

Sending: an SBV_Channel_Close primitive results in sending an SBV_Channel_Close_TC PDU; after which the Access Function is waiting for an SBV_Channel_Close Confirm positive or negative.

Receiving: a valid incoming SBV_Channel_Close_TC PDU results in an SBV_Channel_Close Indication. The Terminal Function closes the requested communication; when the release of the requested communication is terminated the Terminal Function issues a SBV_Channel_Close Response primitive positive which results in sending an SBV_Channel_Close_TC PDU with the appropriate result code. If the Reference parameter is invalid, the Terminal Function issues an SBV_Channel_Close Response primitive negative which results in sending an SBV_Channel_Close_TC PDU with the appropriate Result parameter.

8.3.1.3 SBV_Channel_Error

The protocol elements providing the SBV_Channel_Error service are carried by a Telematic Command as described in table 33 below.

Table 33

Service	Telematic Command	Reference
SBV_Channel_Error	SBV_Channel_Error_TC	9.2 in this ETS

Procedures:

Sending: an SBV_Channel_Error primitive results in sending an SBV_Channel_Error_TC PDU.

Receiving: a valid incoming SBV_Channel_Error_TC PDU results in an SBV_Channel_Error Indication.

8.3.2 Application selection services

8.3.2.1 SBV_Begin_Application

The protocol elements providing the SBV_Begin_Application service are carried by a Telematic Command as described in table 34 below.

Table 34

Service	Telematic Command	Reference
SBV_Begin_Application	SBV_Begin_Application_TC	9.2 in this ETS

Proceures:

Sending: an SBV_Begin_Application primitive results in sending an SBV_Begin_Application_TC PDU; after which the Terminal function is waiting for an SBV_Begin_Application Confirm positive or negative.

Receiving: a valid incoming SBV_Begin_Application_TC PDU results in an SBV_Begin_Application Indication. The Access Function tries to establish the communication with the indicated Application; when the establishment of the requested communication has been terminated (successfully or not) the Access Function issues a SBV_Begin_Application Response primitive which results in sending an SBV_Begin_Application_TC PDU with the appropriate result code.

8.3.2.2 SBV_End_Application

The protocol elements providing the SBV_End_Application service are carried by a Telematic Command as described in table 35 below.

Table 35

Service	Telematic Command	Reference
SBV_End_Application	SBV_End_Application_TC	9.2 in this ETS

Procedures:

Sending: an SBV_End_Application primitive results in sending an SBV_End_Application_TC PDU.

Receiving: a valid incoming SBV_End_Application_TC PDU results in an SBV_End_Application Indication.

8.3.3 Transparent processable data services

8.3.3.1 SBV_TPD_Begin

The protocol elements providing the SBV_TPD_Begin service are carried by a Telematic Command as described in table 36 below.

Table 36

Service	Telematic Command	Reference
SBV_TPD_Begin	SBV_TPD_Begin_TC	9.2 in this ETS

Procedures:

Sending: an SBV_TPD_Begin Request primitive results in sending an SBV_TPD_Begin_TC PDU; after which the sending side is waiting for an SBV_TPD_Begin Confirm positive or negative.

Receiving: a valid incoming SBV_TPD_Begin_TC PDU results in an SBV_TPD_Begin Indication. The receiving side issues a SBV_TPD_Begin Response primitive which results in sending an SBV_TPD_Begin_TC PDU with the appropriate result code. If the Result parameter is set to "TPD not supported" the exchange of processable data is implicitly terminated.

8.3.3.2 SBV_TPD_End

The protocol elements providing the SBV_TPD_End service are carried by a Telematic Command as described in table 37 below.

Table 37

Service	Telematic Command	Reference
SBV_TPD_End	SBV_TPD_End_TC	9.2 in this ETS

Procedures:

Sending: the termination of the transparent processable data exchange may be initiated by both sides at any time by issuing an SBV_TPD_End Request primitive.

Receiving: a valid incoming SBV_TPD_End_TC PDU results in an SBV_TPD_End Indication primitive. A SBV_TPD_End_TC PDU may occur even if there has been no Regime activated by a preceding invocation of the SBV_TPD_Begin Service.

8.3.4 SBV_DFK

The protocol elements providing the SBV_DFK service are carried by a Telematic Command as described in table 38 below.

Table 38

Service	Telematic Command	Reference
SBV_DFK	SBV_DFK_TC	9.2 in this ETS

Procedures:

Sending: a SBV_DFK Request primitive results in an SBV_DFK_TC PDU.

Receiving: a valid incoming SBV_DFK_TC PDU results in an SBV_DFK Indication primitive.

An attempt to manipulate a function key with an identification which is not supported shall result in sending an SBV_TC_Error.

8.3.5 SBV_Escape

The protocol elements providing the SBV_Escape service are carried by a Telematic Command as described in table 39 below.

Table 39

Service	Telematic Command	Reference
SBV_Escape	SBV_Escape_TC	9.2 in this ETS

Procedures:

Sending: a SBV_Escape Request primitive results in an SBV_Escape_TC PDU.

Receiving: a valid incoming SBV_Escape_TC PDU results in an SBV_Escape Indication primitive.

9 Coding

9.1 Introduction

This Clause describes the coding of the parameters which are introduced in this ETS.

Subclause 9.2 defines the coding structure of Telematic Commands and allocates code points for the various protocol elements defined in this ETS.

The coding structure of user data fields is defined in subclause 9.3. This structure provides for both a compact but limited and an extendable coding. This structure is applicable on the User-to-user information (UUI) field of a D-channel SETUP message, the User data fields of CCITT Recommendation X.25 [4] layer 3 packets and the Message field of Telematic Commands.

In subclause 9.4, the mapping of parameters on D-channel messages and parameter fields of PDUs is described.

9.2 Coding of the telematic commands

Telematic Commands are carried in a complete packet sequence of CCITT Recommendation X.25 [4] PLP DATA packets as defined in ISO 8208 [21] with the Q-bit set to 1.

The first octet carried in the User data field of the first packet of this sequence shall be coded 04/00.

NOTE 1: This value is reserved in a future version of draft CCITT Recommendation X.29 (92) and indicates "Telematic service message, Videotex".

Octet 1	04/00	User data field of DATA packet 1
Octet 2	09/14	
Octet 3	TC code	
Octet 4	TC length	
Octet 5	TC message . . .	. .
		. .
Octet 1	TC message (continued) . . .	User data field of DATA packet N . . .

Figure 6: Coding and structure of a Telematic Command

The second octet in the User data field of the first packet shall be coded 09/14, indicating "ISDN syntax-based Videotex Telematic Command".

NOTE 2: Both the protocol defined in this ETS and final draft prETS 300 106 "International Videotex interworking between a terminal and a host" make use of the future draft CCITT Recommendation X.29 (92) extension "Telematic service message, Videotex". The value 09/14 corresponds to "tag [30]" as reserved for the ISDN syntax-based Videotex Terminal protocol in Annex A of final draft prETS 300 106.

The third octet contains the codepoint of the Telematic Command.

The fourth octet carries the total length of the subsequent Telematic Command Message (TCM). This length may be in the range from 0 (00/00) to 254 (15/14). The value 255 (15/15) is reserved for future extensions.

The TCM shall start on the fifth octet in the User data field of the first packet of the sequence. It may be extended to the User data fields of the subsequent packets of the sequence. The TCM carries the parameters of the Telematic Command.

A summary of the structure of a TC is given in figure 6 above.

The range of Telematic Command code (octet 3) values is divided into three parts:

00/00 - 07/14:	used for services defined in this ETS;
08/00 - 11/14:	used for SBV_Escape subservices;
12/00 - 15/15:	for private use.

Both values 07/15 and 11/15 are reserved for future extensions.
The Telematic Command code values are given in table 40 below.

Table 40: Telematic Command code values

Telematic Command	Telematic Command code
SBV_TC_Error_TC	00/01
SBV_Channel_Open_TC	00/02
SBV_Channel_Close_TC	00/03
SBV_Channel_Error_TC	00/04
SBV_TFI_TC	00/05
SBV_Begin_Application_TC	00/06
SBV_End_Application_TC	00/07
SBV_TPD_Begin_TC	00/08
SBV_TPD_End_TC	00/09
SBV_DFK_TC	00/10
NOTE: Octet 3 can directly indicate the purpose of the command. In that case, it corresponds to the Service-ID parameter of the SBV_Escape service (cf. Annex D (informative)).	

9.3 Coding structures

The coding of the various parameters given in table 44 (cf. subclause 9.4.4) uses a Type, Length, Value (TLV) method. This coding offers a basic structure and an extended structure.

NOTE : The Basic Coding Structure is necessary due to the limited amount of data of the User data fields of the CALL REQUEST and INCOMING CALL packets when the CCITT Recommendation X.25 [4] Fast Select facility is not used. However, to encode the different parameters carried in one service element, it may be necessary to employ both coding methods, respectively.

When applying this coding scheme to the User data field of the CALL REQUEST/INCOMING CALL packet (regardless whether the CCITT Recommendation X.25 [4] Fast Select facility is used or not), the coding shall start on the fifth octet of the User data field. CCITT Recommendation X.29 [5] reserves the first four octets for the protocol identifier (cf. final draft prETS 300 080 [15], §7.2.8 and CCITT Recommendation X.29 [5], §4.2).

9.3.1 Basic coding structure

In this coding structure, the type indicator and the length indicator are coded on 4 bits, respectively. The value of the length indicates the length in octets of the following field. Figure 7 gives an overview:

	MSB	LSB	
Octet 1	Type 1	Length 1	Length between 0 and 14
Octet 2 . .	Value 1 . .		
Octet N	Type M	Length M	Length between 0 and 14
Octet N+1 . .	Value M . .		

Figure 7: Basic coding structure

The type indicator may take the values:

0 - 7:	not defined in this ETS;
8 - 14:	defined in this ETS;
15 :	reserved for the extended coding structure.

The length indicator may take values between 0 and 14.

A value of 15 in the length indicator implies that the following octet represents the actual length of the parameter value. This actual length shall take a value in the range from 15 (00/15) to 127 (07/15). Values below 15 (00/15) or above 127 (07/15) are reserved for future extensions.

Figure 8 predicts the coding scheme for that case:

	MSB	LSB	
. .	. .		
Octet N	Type M	15	
Octet N+1	Length M		Length between 15 and 127
Octet N+2 . .	Value M . .		

Figure 8: Basic coding structure with length extension mechanism

9.3.2 Extended coding structure

If the first octet of a parameter carries the value 255 (15/15), the following octets shall contain an extended type indicator coded in one octet and a length indicator coded in one octet, as presented in figure 9.

	MSB LSB	
. .	. .	
Octet N	15/15	
Octet N+1	Extended Type M	
Octet N+2	Length M	Length between 0 and 127
Octet N+3 . .	Value M . .	

Figure 9: Extended coding structure

The extended type indicator may take the values:

00/00 - 07/15:	not defined in this ETS;
08/00 - 15/15:	defined in this ETS.

NOTE: In order to avoid any ambiguities between the basic and the extended coding structures, type indicator values x/00 (x in the range from 8 to 15) are reserved for use with the basic coding structure.

In the extended coding structure, the length indicator may take a value in the range from 0 (00/00) to 127 (07/15). Values greater than 127 are reserved for future extensions.

9.3.3 Coding of data types

9.3.3.1 Integers

Integer values shall be coded in one octet. The contents of this octet shall be the binary number equal to the integer value.

NOTE: Integers coded in more than one octet are for further study.

9.3.3.2 Strings

String values shall be coded as a sequence of characters as defined in CCITT Recommendation T.51 [2].

9.3.3.3 Voids

The void data type shall be coded having zero-length and no value.

9.3.3.4 Octet sequences

The octet sequence shall be coded as a sequence of arbitrary octets.

9.4 Coding of parameters

9.4.1 Coding of parameter fields as defined in other documents

This clause describes parameters which are mapped on fields other than User data fields. Some parameters introduced in this ETS are directly mapped onto layer 3 PDU parameter fields. In such a case, the coding of the parameters shall be defined in the various Standards or Recommendation as defined in table 41 below.

Table 41

Parameter	PDU(s)	Parameter field name	Standard/Recommendation
OB_Called_Address	SETUP	Called party number	ETS 300 102-1 [16] §4.5.8
OB_Called_Subaddress	SETUP	Called party subnumber	ETS 300 102-1 [16] §4.5.9
OB_Cause	DISCONNECT DISCONNECT/RELEASE RELEASE COMPLETE	Cause Cause Cause	ETS 300 102-1 [16] §4.4 ETS 300 102-1 [16] §4.4 ETS 300 102-1 [16] §4.4
IB_Called_Address	CALL REQUEST INCOMING CALL	Called DTE Address Called DTE Address	ISO 8208 [21] §12.2.1.1.3 ISO 8208 [21] §12.2.1.1.3
IB_Cause	CLEAR REQUEST CLEAR INDICATION CLEAR CONFIRMATION	Clearing Cause Clearing Cause Clearing Cause	ISO 8208 [21] §12.2.3.1.1 ISO 8208 [21] §12.2.3.1.1 ISO 8208 [21] §12.2.3.1.1
IB_Diagnostic	CLEAR REQUEST CLEAR INDICATION CLEAR CONFIRMATION	Diagnostic Code Diagnostic Code Diagnostic Code	ISO 8208 [21] §12.2.3.1.2 ISO 8208 [21] §12.2.3.1.2 ISO 8208 [21] §12.2.3.1.2
X.3_Parameter_List	X.29 PAD message	Data	X.29 [5] §4.4.5

9.4.2 Coding of User data fields as defined in other documents

Table 42 shows parameters which are carried in the user data fields of CCITT Recommendation X.25 [4] PDUs or the Telematic Command Message in Telematic Commands. The coding of these parameters shall be performed according to the various Standards as defined in table 42, without using the coding structure as defined in subclause 9.3.

Table 42

Parameter	Message/PDU(s)	Standard	Remarks
VTX_Data	DATA (UD)	ETS 300 072 to 300 076 ([10] to [14]) ETS 300 177 [19] ETS 300 149 [18]	NOTE 1
TFI_Enq	SBV_TFI_TC (TCM)	ETS 300 076 [14]	NOTE 2 and NOTE 3
TFI_Ack	SBV_TFI_TC (TCM)	ETS 300 076 [14]	NOTE 2 and NOTE 3
Abbreviations: UD User Data field in a DATA packet TCM Telematic Command Message in a Telematic Command			
NOTE 1: The VTX_Data parameter is directly mapped on the User data field of a DATA packet and coded in accordance to ETS 300 072 to 300 076 ([10] to [14]), ETS 300 177 [19], and ETS 300 149 [18]; it may therefore contain a TFI request (and response) in accordance to ETS 300 076 [14]. If used in conjunction with the SBV_TPD services, this parameter may also carry processable data without making use of the DDU layer as specified in ETS 300 075 [13].			
NOTE 2: These parameters are directly mapped onto the Telematic Command Message of the SBV_TFI TC and coded in accordance with ETS 300 076 [14]; the support of the particular Videotex presentation data element (VPDE) request is optional.			
NOTE 3: The application of future CCITT-defined recommendations is for further study.			

9.4.3 Coding of user data fields using free format

Table 43 shows parameters which are carried in the User data fields of CCITT Recommendation X.25 [4] PDUs or the Telematic Command Message in Telematic Commands. The coding of these parameters is not described in this ETS.

NOTE: To permit the distinction of a free-formatted parameter from parameters coded in accordance to subclause 9.3, it is required to set the MSB of the first octet of the free-formatted parameter to 0.

Table 43

Parameter	Message/PDU(s)	Remarks
OB_User_Data	SETUP (UUI)	NOTE 1
IB_User_Data	CALL REQUEST (CUD) INCOMING CALL (CUD)	NOTE 1 and NOTE 2
IB_User_Data	SBV_Begin_Application_TC (TCM)	NOTE 1
Abbreviations: UUI User to User Information field in the SETUP message CUD Call User Data field in the CALL REQUEST/INCOMING CALL packets TCM Telematic Command Message in a Telematic Command		
NOTE 1: If used in conjunction with other parameters which are coded in accordance to subclause 9.3 of this ETS, this parameter shall appear as the last one in the User data field.		
NOTE 2: The coding shall start on the fifth octet of the User data field. The first four octets are reserved by CCITT Recommendation X.29 [5].		

9.4.4 Coding of user data fields using the coding structures

Table 44 defines the coding of parameters which are carried in the User-to-User information field in the SETUP message, the User data fields in CCITT Recommendation X.25 [4] PDUs or the Telematic Command Message in Telematic Commands, using the coding structure as defined in subclause 9.3.

Table 44

Parameter	Message/PDU(s)	Type BCS	Type ECS	Value or Remarks
Req_OB_Called_Address	SBV_Channel_Open_TC (TCM)	--	08/12	NOTE 1
Req_OB_Called_Subaddress	SBV_Channel_Open_TC (TCM)	--	08/13	NOTE 1
OB_Application_Address	SETUP (UUI) SBV_Channel_Open_TC (TCM)	--	09/08	NOTE 2
OB_Application_Selection	SETUP (UUI) SBV_Channel_Open_TC (TCM)	--	10/08	Not defined in this ETS
OB_Application_Data	SETUP (UUI) SBV_Channel_Open_TC (TCM)	--	11/08	Not defined in this ETS
Req_OB_User_Data	SBV_Channel_Open_TC (TCM)	--	08/06	Free format
Ind_OB_Cause	SBV_Channel_Open_TC (TCM) SBV_Channel_Error_TC (TCM)	--	08/11	NOTE 3
Req_IB_Called_Address	SBV_Channel_Open_TC (TCM)	--	08/04	NOTE 4
IB_Application_Address	CALL REQUEST/ INCOMING CALL (CUD) SBV_Channel_Open_TC (TCM) SBV_Begin_Application_TC (TCM)	9	--	NOTE 2
IB_Application_Selection	CALL REQUEST/ INCOMING CALL (CUD) SBV_Channel_Open_TC (TCM) SBV_Begin_Application_TC (TCM)	10	--	Not defined in this ETS
IB_Application_Data	CALL REQUEST/ INCOMING CALL (CUD) SBV_Channel_Open_TC (TCM) SBV_Begin_Application_TC (TCM)	11	--	Not defined in this ETS
Req_IB_User_Data	SBV_Channel_Open_TC (TCM)	--	08/07	Free format

Table 44 (concluded)

Packet_Size	SBV_Channel_Open_TC (TCM)	--	13/01	Integer (NOTE 5) 7 ::= 128 octets 8 ::= 256 octets 9 ::= 512 octets 10 ::= 1024 octets 11 ::= 2048 octets
Parameter	Message/PDU(s)	Type BCS	Type ECS	Value or Remarks
Window_Size	SBV_Channel_Open_TC (TCM)	--	13/02	Integer (NOTE 5) 1..7 ::= Window size 1..7
Ind_IB_Cause	SBV_Channel_Open_TC (TCM) SBV_Channel_Error_TC (TCM)	--	08/03	NOTE 3
Ind_IB_Diagnostic	SBV_Channel_Open_TC (TCM) SBV_Channel_Error_TC (TCM)	--	08/05	NOTE 3
Reference	SBV_Channel_Open_TC (TCM) SBV_Channel_Close_TC (TCM) SBV_Channel_Error_TC (TCM) CALL REQUEST/ INCOMING CALL (CUD)	12	--	Not defined in this ETS
Result	SBV_Channel_Open_TC (TCM) SBV_Channel_Close_TC (TCM) SBV_Begin_Application_TC (TCM) SBV_TPD_Begin_TC (TCM) CALL ACCEPTED/ CALL CONNECTED (CUD) CLEAR REQUEST/ CLEAR INDICATION (CUD)	13	--	Integer 0 ::= "Successful" 1 ::= "Not successful" 2 ::= "Application not available" 3 ::= "Application already engaged" 4 ::= "Release not allowed" 5 ::= "TPD not supported"
Reverse_Role_Assignment	CALL REQUEST/ INCOMING CALL (CUD)	14	--	Void
Error_Code	SBV_TC_Error_TC (TCM)	--	08/01	Octet Sequence (cf. 9.4.5.1)
DDU_Fall-Back_Mode	SBV_TPD_Begin_TC (TCM)	--	12/02	String (cf. 9.4.5.2)
Function_Keys	SBV_DFK_TC (TCM)	--	14/01	Octet Sequence (cf. 9.4.5.3)
Reset_Keys	SBV_DFK_TC (TCM)	--	14/02	Void (cf. 9.4.5.3)

Abbreviations:

BCS	Basic Coding Structure
ECS	Extended Coding Structure
TCM	Telematic Command Message in a Telematic Command
UUI	User to User Information field in the SETUP message
CUD	Call User Data field in the CALL REQUEST/INCOMING CALL packets
or	Called User Data field in the CALL ACCEPTED/CALL CONNECTED packets (may only be used in conjunction with the Fast Select facility)
or	Clear User Data field in the CLEAR REQUEST/CLEAR INDICATION packets (may only be used in conjunction with the Fast Select facility)

NOTE 1: These parameters can be mapped onto the OB_Called_Address and the OB_Called_Subaddress parameters of the SBV_Establish service, respectively. Therefore, they shall be coded in the same way as these parameters (cf. subclause 9.4.1).

NOTE 2: It is recommended to apply CCITT Recommendation X.121 [7].

NOTE 3: These parameters carry the same type of information as the OB_Cause, IB_Cause and IB_Diagnostic parameters, respectively. Therefore, they shall be coded in the same way as these parameters (cf. subclause 9.4.1).

NOTE 4: This parameter can be mapped onto the IB_Called_Address parameter of the SBV_Establish service. Therefore, it shall be coded in the same way as the IB_Called_Address parameter (cf. subclause 9.4.1).

NOTE 5: Other values are for further study.

NOTE 6: The Error_Detection_Location parameter is local only. Therefore, no value is given in this table. Nevertheless, value 08/02 shall be reserved with respect to future extensions, if any.

9.4.5 Service specific coding

9.4.5.1 Coding of the Error_Code parameter

The Error_Code parameter carried in the SBV_TC_Error TC is coded as a sequence of octets; the first octet (coded as an Integer) may take the value:

00/00: "TC not supported";
00/01: "TC erroneous".

In the first case ("TC not supported"), the following octet shall contain the TC code of the received TC indication which is not supported.

EXAMPLE 1: This example shows the complete coding of the SBV_TC_Error TC for the case "TC not supported". It is assumed that the SBV_TPD_Begin service is not implemented.

Octet 1	04/00	Telematic service message
Octet 2	09/14	ISDN syntax-based Videotex
Octet 3	00/01	SBV_TC_Error_TC
Octet 4	00/05	Length of TCM: 5 octets
Octet 5	15/15	Extended coding structure
Octet 6	08/01	Type: Error_Code
Octet 7	00/02	Length: 2 octets
Octet 8	00/00	Value: "TC not supported"
Octet 9	00/08	Value: SBV_TPD_Begin

Figure 10: Example 1 for the coding of the Error_Code parameter

The second case ("TC erroneous") occurs if the TC is supported, but the coding of one or several of the associated parameters is not correct.

In that case, the following octets shall contain the TC code of the received TC indication, followed by a list of the erroneous parameter(s). This list shall consist of octets containing the separate type indicator(s) of the erroneous parameter(s) involved. Type indicators which are defined for the Basic Coding Structure (cf. subclause 9.3.1) shall be coded in the upper four bits of an octet, with the lower four bits set to 0.

EXAMPLE 2: The following example shows the complete coding of the SBV_TC_Error TC for the case "TC erroneous". It is assumed that the minimum set of the SBV_DFK service (i.e., 8 keys) is implemented, but a previously received SBV_DFK indication has tried to manipulate function key 10.

Octet 1	04/00	Telematic service message
Octet 2	09/14	ISDN syntax-based Videotex
Octet 3	00/01	SBV_TC_Error_TC
Octet 4	00/06	Length of TCM: 6 octets
Octet 5	15/15	Extended coding structure
Octet 6	08/01	Type: Error_Code
Octet 7	00/03	Length: 3 octets
Octet 8	00/01	Value: "TC erroneous"
Octet 9	00/10	Value: SBV_DFK
Octet 10	14/01	Value: Function_Keys

Figure 11: Example 2 for the coding of the Error_Code parameter

9.4.5.2 Coding of the DDU_Fall-Back_Mode parameter

The DDU_Fall-Back_Mode parameter is coded as a string.

The possible characters in the string are restricted to:

"A","B","C","D","E","F" and "G" (uppercase only).

The string may contain one or several characters; the characters shall follow the alphabetic order; a character shall not be duplicated.

EXAMPLE: Some possible valid values of the DDU_Fall-Back_Mode parameter:
- "B";
- "ABDEG";
- "ABCDEFG".

9.4.5.3 Coding of the Function_Keys parameter

The SBV_DFK TC Function_Keys parameter is a structured data type. It carries a list of function key definitions. Each function key consists of an identification, an optional user visible name, an optional code sequence and an optional "do-not-forward" indication.

The user visible name may be used by the Terminal Function to inform the user about the purpose of a function key. The code sequence is a sequence of octets which has to be sent to the Access Function when the function key is depressed. By default, the function key shall be associated with a forwarding condition. The presence of a do-not-forward indication defines that the function key shall not be associated with a forwarding condition.

The Terminal Function shall process the conditions in the received order. When setting a specific function key twice in one Telematic Command, the value of the last received definition shall be used. Reset commands which are sent in between function key lists shall be processed.

The list of function keys is coded as a structure. Each element of this structure is coded in a TLV (type-length-value) form, which conforms to the extended coding structure as defined in subclause 9.3.2.

The identification is coded as an integer. For terminals supporting the SBV_DFK primitive, at least values 1 to 8 shall be supported. The user visible name shall be encoded as a string of the indicated length. The code sequence shall be encoded as a sequence of octets of the indicated length. The values of these octets are not limited to a specific data syntax. The do-not-forward indication shall be encoded as a void type.

Table 45 defines the coding of the type indicators:

Table 45

Type indicator	Data element
06/00	Function Key
06/01	Identification
06/02	User Visible Name
06/03	Code Sequence
06/04	Do-not-forward

The following formal specification gives the syntax of the Function_Keys data-structure:

```
Function_Keys          ::=    "15/15" "14/01" Length Function_Key_List

Function_Key_List      ::=    Function_Key        Function_Key_List
                       |      /* empty */

Function_Key           ::=    "06/00"     Length       Identification
                                          User_visible_name
                                          Code_sequence
                                          Do_not_forward

Identification         ::=    "06/01"     Length Integer

User_visible_name      ::=    "06/02"     Length String
                       |      /* empty */

Code_sequence          ::=    "06/03"     Length Octet_sequence
                       |      /* empty */

Do_not_forward         ::=    "06/04" "00/00"
                       |      /* empty */

Length                 ::=    "Octet conforming to the length type in 9.3.2"
Integer                ::=    "Integer conforming to 9.3.3.1"
String                 ::=    "String conforming to 9.3.3.2"
Octet_sequence         ::=    "Octet sequence conforming to 9.3.3.4"
```

The following examples illustrate the use of the SBV_DFK service, showing the complete coding for the SBV_DFK TC:

EXAMPLE 1: After deleting all current definitions, associate key 1 with the user visible name "F1" and the code string "Code1", and key 10 with the user visible name "F10" and the code sequence "Code2":

04/00	Telematic service message		
09/14	ISDN syntax-based Videotex		
00/10	SBV_DFK_TC		
02/07	Length of TCM: 39 octets		
15/15	Extended coding structure		
14/02	Reset		
00/00	length = 0		
15/15	Extended coding structure		
14/01	function key list		
02/01	length = 33		
	06/00	function key	
	00/14	length = 14	
		06/01	identification
		00/01	length = 1
		00/01	value = 1
		06/02	user visible name
		00/02	length = 2
		"F"	value = "F1"
		"1"	
		06/03	code sequence
		00/05	length = 5
		"C"	value = "Code1"
		"o"	
		"d"	
		"e"	
		"1"	
	06/00	function key	
	00/15	length = 15	
		06/01	identification
		00/01	length = 1
		00/10	value = 10
		06/02	user visible name
		00/03	length = 3
		"F"	value = "F10"
		"1"	
		"0"	
		06/03	code sequence
		00/05	length = 5
		"C"	value = "Code2"
		"o"	
		"d"	
		"e"	
		"2"	

Figure 12: Example 1 for the coding of the Function_Keys parameter

EXAMPLE 2: Erase reset key 10 and overwrite key 1 with "FN" for the user visible name and code string "New", which is not a forwarding condition:

04/00	Telematic service message		
09/14	ISDN syntax-based Videotex		
00/10	SBV_DFK_TC		
01/08	Length of TCM: 24 octets		
15/15	Extended coding structure		
14/01	function key list		
01/05	length = 21		
	06/00	function key	
	00/03	length = 3	
		06/01	identification
		00/01	length = 1
		00/10	value = 10
	06/00	function key	
	00/14	length = 14	
		06/01	identification key
		00/01	length = 1
		00/01	value = 1
		06/02	user visible name
		00/02	length = 2
		"F"	value = "FN"
		"N"	
		06/03	Code sequence
		00/03	length = 3
		"N"	value = "New"
		"e"	
		"w"	
		06/04	Do not_forward
		00/00	length = 0

Figure 13: Example 2 for the coding of the Function_Keys parameter

10 Use of CCITT Recommendation X.3 [3] parameters

CCITT Recommendation X.3 [3] parameters shall be used either as described in this ETS or as described in the relevant CCITT Recommendations. The values of the CCITT Recommendation X.3 [3] parameters given in this ETS shall be changed only through the use of the SBV_Set_Param and SBV_Set_Read_Param services. The values shall not be changed by the Terminal Function itself.

10.1 Selectable functions provided by ISDN syntax-based Videotex Terminals

NOTE: For a global description of these functions, refer to the Terminal Model described in subclause 6.2 of this ETS.

10.1.1 Standard functions

For a description of the "Echo", "Selection of the data forwarding characters", and "Selection of idle timer delay" functions refer also to CCITT Recommendation X.3 [3], §1.4.

10.1.2 Extended functions

10.1.2.1 Size of input field

This function provides for a maximum length definition of an input field.

When this parameter is not implemented or the input field length is set to 0 none of the additional rules given below shall apply.

A non-zero input field length implicitly defines an input field which is used for user input collection inside the Terminal Function. The input field length is counted in terms of graphic characters (cf. ETS 300 072 [10], subclause 1.1.1).

All forwarding conditions shall apply except for the packet full condition (cf. subclause 6.2 of this ETS). In addition, the user input shall be forwarded in a complete packet sequence whenever as many characters as indicated by the parameter value are collected in the input field.

10.1.2.2 End-of-frame signals

This function enables the terminal to recognise the end of a frame.

NOTE: The definition of a frame is outside the scope of this ETS.

End-of-frame signals may be used to indicate when the terminal can start the echoplex procedure (cf. subclause 6.2 of this ETS).

10.1.2.3 Extended data forwarding signals

This function extends the selection of data forwarding characters by some videotex service specific values. A clarification of the use of these extended data forwarding signals is given in Annex E (informative).

10.2 List of CCITT Recommendation X.3 [3] parameters and possible values

10.2.1 Standard parameters

For a list of possible values for parameter references 2, 3, 4 and 11 refer to subclause 10.2.3, table 46.

10.2.2 Extended parameters

TEMPORARY NOTE: Agreement was reached within CCITT SGVII (November 1990) to incorporate the following parameter references into the next revised version of CCITT Recommendation X.3 [3]. Until then, the following parameter references are considered "private" in the sense of CCITT Recommendation X.29 [5], §4.4.5.4.

10.2.2.1 Size of input field

Parameter reference 23

The following values may be used:

no input field length	- represented by decimal 0;
input field length	- represented by decimal 1-255.

10.2.2.2 End-of-frame signals

Parameter reference 24

The parameter is represented by the following encoding of basic functions, each having a decimal value as shown below:

no end-of-frame signal	- represented by decimal 0;
the end-of-frame condition is fulfilled upon receipt of this character	- represented by decimal 1-31;
the end-of-frame condition is fulfilled upon receipt of a complete packet sequence	- represented by decimal 32;
the end-of-frame condition is fulfilled upon detection of the end of a timer	- represented by decimal 64.

NOTE: The decimal representation of individual values of this parameter allows for a coding to represent a single function or a combination of functions.

10.2.2.3 Extended data forwarding signals

Parameter reference 25

The parameter is represented by the following encoding of basic functions, each having a decimal value as shown below:

no extended data forwarding condition	- represented by decimal 0;
Sequence of two characters starting with character 1/3	- represented by decimal 1;
Character 1/10	- represented by decimal 2;
Character 1/12	- represented by decimal 4;
Character 2/3	- represented by decimal 8;
Character 5/15	- represented by decimal 16;
Sequence of the two characters ESC "J" (1/11 4/10)	- represented by decimal 32;
Sequence of three characters starting with the two-character sequence 2/10 3/0	- represented by decimal 64.

NOTE: The decimal representation of individual values of this parameter allows for a coding to represent a single function or a combination of functions.

10.2.3 Overview of supported CCITT Recommendation X.3 [3] parameters

Table 46: Possible values and combinations of values of supported CCITT Recommendation X.3 [3] parameters

Parameter reference number	Parameter description	Mandatory values	Optional values	X.3 parameter meaning	Remarks
2	Echo (M)	0		No echo	
		1		Echo	
			32 to 126	Scrambled echo character	NOTE 1
3	Selection of data forwarding characters (M)	0		No data forwarding characters	
		1 (NOTE 2)		Alphanumeric characters (A-Z,a-z,0-9)	
		2		Character CR	
			4	Characters ESC, BEL, ENQ, ACK	
			8	Characters DEL, CAN, DC2	
		16 (NOTE 2)		Characters ETX, EOT	
			32	Characters HT, LF, VT, FF	
			64	All Characters in columns 0 and 1 of IA5 except those shown above for values 2, 4, 8, 16, 32	
		126		All characters in columns 0 and 1 and character DEL	
		(NOTE 3)	(NOTE 3)		
4	Selection of idle timer delay (M)	0			NOTE 4
		1 (NOTE 5)		Value of idle timer in twentieths of a second	
			2 to 19		
		20			
			21 to 254		
		255			
11 (read only)	Binary speed (M)	18		64000 bit/s	NOTE 6
23	Size of input field (O) (TEMP.NOTE 1)	0		Undefined size (NOTE 7)	TEMP.NOTE 2
			1 to 255	Length of input field (number of character positions)	
24	End-of-frame signals (O) (TEMP.NOTE 1)	0		No end-of-frame signal	
			1 to 31	Character representing the end of a frame	
		32		A frame is represented by a complete packet sequence	
			64	At the end of a timer, an end-of-frame condition may be assumed	
			(NOTE 3)	(NOTE 8)	
25	Extended data forwarding signals (O) (TEMP.NOTE 1)	0		No extended data forwarding signal	TEMP.NOTE 3
			1	Sequence of two characters 1/3 x/y (NOTE 9)	
			2	Character 1/10	
			4	Character 1/12	
			8	Character 2/3	
			16	Character 5/15	
			32	Sequence of the two characters ESC "J" (1/11 4/10)	
			64	Sequence of three characters 2/10 3/0 x/y	
			(NOTE 3)		

Table 46 (concluded)

Abbreviations:

M	Mandatory
O	Optional

NOTE 1: These parameter values are an extension of CCITT Recommendation X.3 [3].

NOTE 2: This parameter value is optional in CCITT Recommendation X.3 [3].

NOTE 3: These parameter values may be combined.

NOTE 4: If an optional parameter value is requested, the nearest locally known value may be used.

NOTE 5: This parameter value is optional in CCITT Recommendation X.3 [3]. Some ISDN syntax-based Videotex Terminals may interpret this value in the sense of "as soon as possible".

NOTE 6: This is a "misuse" of the original parameter reference 11 as defined in CCITT Recommendation X.3 [3]. The use of other values (e.g. DTE/DCE communication over the D-channel) is for further study.

NOTE 7: If no size of an input field is defined, only forwarding conditions defined by parameter reference 3 and parameter reference 25 shall apply.

NOTE 8: The timer value is service-dependent and shall be in the range of 1 to 5 seconds, with a typical value of 3 seconds.

NOTE 9: "x/y" stands for any character.

TEMPORARY NOTE 1: Agreement was reached within CCITT SGVII (September 1990) to incorporate this parameter reference into the next revised version of CCITT Recommendation X.3 [3]. Until then, these parameter references are considered "private" in the sense of CCITT Recommendation X.29 [5], §4.4.5.4.

TEMPORARY NOTE 2: In a liaison document from CCITT SGVII to CCITT SGVIII, the length of an input buffer is defined. It is assumed that this shall be aligned.

TEMPORARY NOTE 3: In a liaison document from CCITT SGVII to CCITT SGVIII, this list is defined slightly different. It is assumed that this shall be aligned.

10.2.4 Other CCITT Recommendation X.3 [3] parameters

If an ISDN syntax-based Videotex Terminal receives a parameter reference which is not contained in the above table 46, it may react as follows:

- The parameter reference shall be processed but otherwise ignored. The referenced value can be set and read, although the associated function is not used in ISDN syntax-based Videotex.

- The terminal answers with a CCITT Recommendation X.29 [5] Parameter indication PAD message (cf. CCITT Recommendation X.29 [5], §4.4.5).

NOTE: To assure the communication with some Videotex Services, it may be imperative to implement the first of the two above solutions.

11 Lower Layers

This clause describes the lower layers (layer 1 to layer 3) for an ISDN syntax-based Videotex Terminal. The description is limited to demand circuit-switched calls using the 64 kbit/s unrestricted digital information bearer capability and the DTE/DTE case of the layer 3 peer entities in the B-channel connection.

NOTE: The lower layer descriptions for demand circuit-switched calls using the DTE/DCE case of the layer 3 peer entities in the B-channel (case A of ETS 300 007) and for calls using the ISDN packet mode for access via the B-channel or via the D-channel (case B of ETS 300 007) are covered in DE/TE-01012.

11.1 Co-ordination between D-channel and B-channel

NOTE: As this ETS cannot make reference to the Connection Oriented Network Service (CONS), it is not possible to base a description of the co-ordination function on final draft prETS 300 080 [15].

11.2 Layer 3 D-channel protocol

11.2.1 The access protocol

Subclause 7.1 of final draft prETS 300 080 [15] shall apply without any additional rule.

11.2.2 Terminal selection and compatibility checking

Subclause 7.2 of final draft prETS 300 080 [15] shall apply without any additional rule.

11.2.3 Service specific use of supplementary services

Subclause 7.3 of final draft prETS 300 080 [15] shall apply. In addition, the following supplementary services may be used optionally:

- Subaddressing (SUB);
- User-to-User Signalling (UUS) Service 1;
- Call Deflection (CD) in combination with UUS Service 1.

11.2.4 Call Progress signals

Call Progress Signals may be handled locally by the terminal.

11.3 Layer 3 B-channel protocol

Subclause 8.2 of final draft prETS 300 080 [15] shall apply with the following additional rules:

- the mapping in accordance with ISO/IEC 8878 does not apply for ISDN syntax-based Videotex.
- Subclause 8.2.11 of final draft prETS 300 080 [15] "Encoding of (NSAP) Addresses" does not apply.

11.4 Layer 2 D-channel protocol

Clause 6 of final draft prETS 300 080 [15] shall apply without any additional rule.

11.5 Layer 2 B-channel protocol

Subclause 8.1 of final draft prETS 300 080 [15] shall apply without any additional rule.

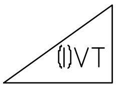
11.6 Layer 1 protocol

Clause 5 of final draft prETS 300 080 [15] shall apply without any additional rule.

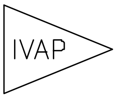
Annex A (informative): Examples of configurations

A.1 Symbols

The following symbols are used in this annex:



(ISDN syntax-based)
Videotex Terminal



ISDN syntax-based
Videotex Access Point

Videotex
Host

A.2 Access to PSTN/PSPDN systems using a terminal adaptor

These configurations are intended to access Videotex Services (VS), designed for use on the PSTN, via an ISDN. The services offered to the user should be identical to the service offered via a PSTN/PSPDN. As a consequence, the protocols to be used at the terminal side are those used for the access of PSTN/PSPDN Videotex Services.

These configurations are outside the scope of this ETS.

A.2.1 Analogue terminal adaptors

An ISDN may be utilized as an analogue telephone network (via the 3,1 kHz audio bearer service) using analogue terminal adaptors (TA). Both sides require modems. As shown in figure A.1, the interworking unit between ISDN and PSTN can be used.

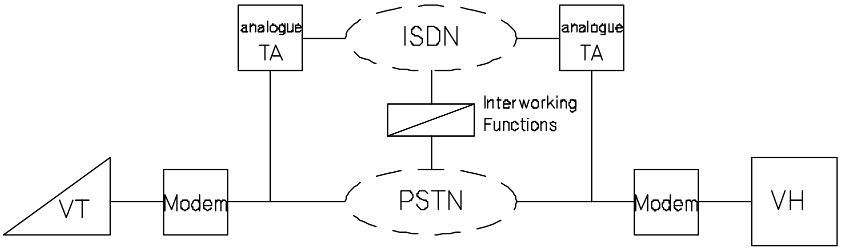


Figure A.1: Access of PSTN/PSPDN systems via analogue terminal Adaptors

A.2.2 Rate adapting terminal adaptors

Rate adapting terminal adaptors provide an end-to-end digital connection which transforms the digital 64 kbit/s link into an asynchronous V.24 link. No modems are needed, as shown in figure A.2.

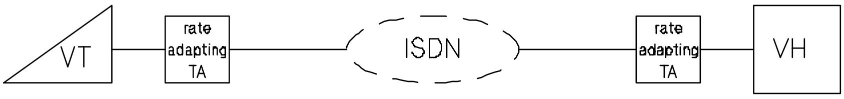


Figure A.2: Access of PSTN/PSPDN systems via rate adapting terminal adaptors

A.3 Connection to a VS

The ISDN Videotex Terminal is connected directly to a Videotex Service via the ISDN. At the protocol level, one connection is established between the terminal and Videotex Service.

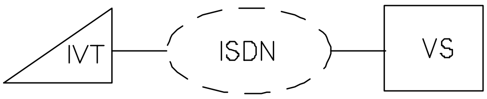


Figure A.3: Connection to a VS

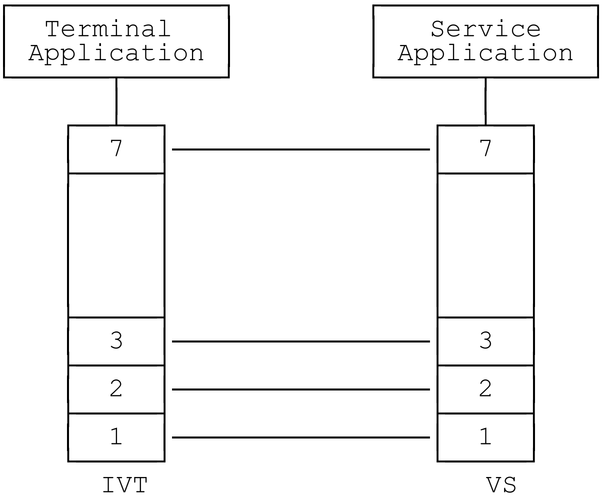
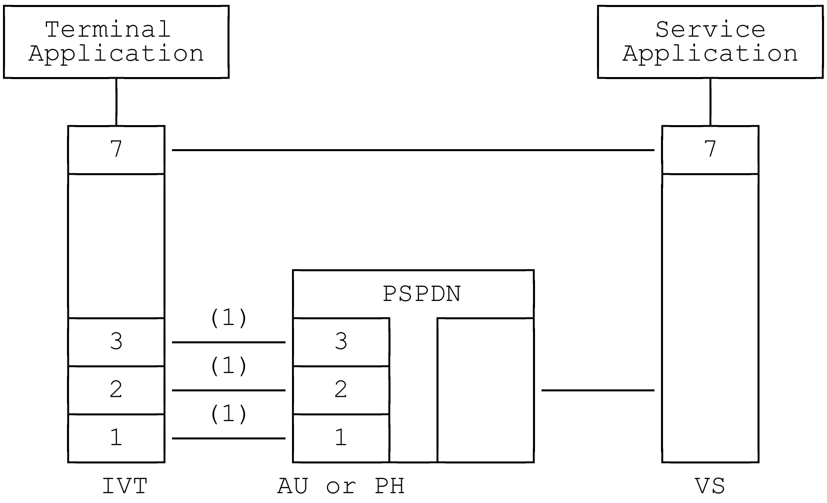


Figure A.4: Protocol stack for IVT-VS connection

A.4 Access to a VS via a PSPDN

A Videotex Service is made available over the ISDN, using the interworking mechanisms between the ISDN and the PSPDN as defined in ETS 300 007 (CCITT Recommendation X.31) for either case A or case B. These topologies imply that the packet interworking unit (an Access Unit (AU) in case A, a Packet Handler (PH) in case B) is addressed over the ISDN and the B-channel (or optionally the D-channel for case B) is used to establish a connection with the Videotex Service.



(1) ETS 300 007 (CCITT Rec. X.31) access

Figure A.5: Protocol stack for IVT-AU/PH-VS connection

A.5 Access to a VS via an IVAP

The terminal accesses a Videotex Service via an ISDN Videotex Access Point. The Host Access Network should be a PDN (generally a PSPDN) or an ISDN.

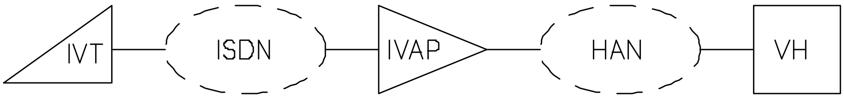


Figure A.6: Connection via an IVAP

The connection establishment with the host does not have any influence on the protocol stack between the terminal and the IVAP. As far as the terminal is concerned, only a connection to the IVAP exists.

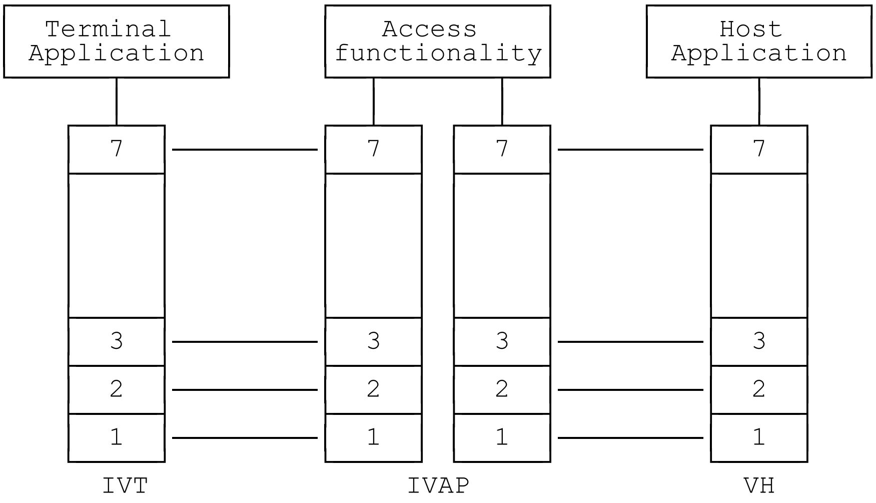


Figure A.7: Protocol stack for IVT-IVAP-VH connection

In terms of the protocol stack in the B-channel, there is an end-to-end connection on layer 3 between the two sides, with an Application Layer on top.

A.5.1 Service selection after a dialogue with the IVAP

The terminal is connected to an IVAP, which sets up a second connection to a server after some dialogue between the terminal and the IVAP. This is basically a refinement of the access to a VH.

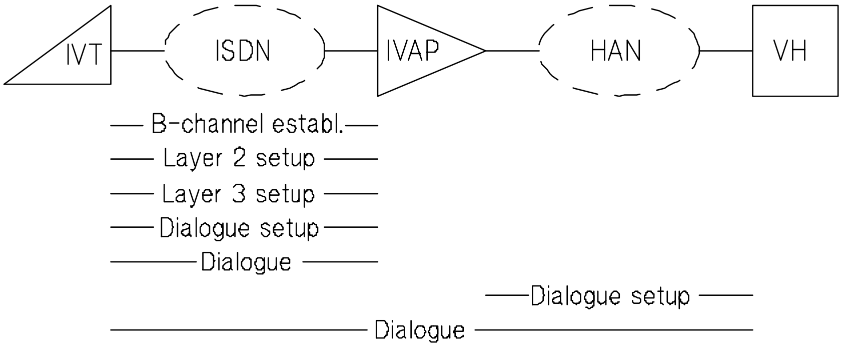


Figure A.8: Connection establishment after dialogue

A.5.2 Service selection using a VS identification

A B-channel is established between the terminal and the IVAP. On this B-channel, layer 2 and layer 3 are set up. The layer 3 CALL REQUEST carries the address or the name of the requested application. This information is used by the IVAP to establish an end-to-end dialogue between the terminal and the VH.

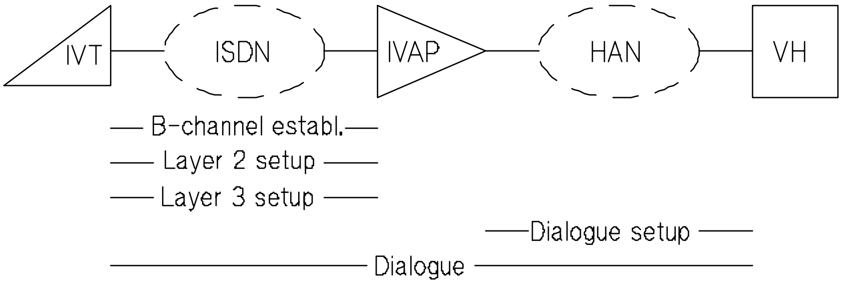


Figure A.9: Connection establishment without dialogue

A.5.2.1 VS identified by a network address

The application is selected by its VH-address on the Host Access Network. This address is inserted in the layer 3 CALL REQUEST and interpreted by the IVAP to establish a connection with the VH.

EXAMPLE: Assume that the VH-address is "12345678". This number is inserted in the layer 3 CALL REQUEST sent by the terminal to the VAP.

NOTE: Some user data (e.g., user-identification) may be associated with the VH-address. It is conveyed transparently from the terminal to the VH.

A.5.2.2 VS identified by a name

The requested application is identified by a Videotex Service name. This name is inserted by the terminal in the layer 3 CALL REQUEST and interpreted by the IVAP to establish the appropriate connection.

EXAMPLE: Assume that the Videotex service name is "ETSI". This name is inserted in the layer 3 CALL REQUEST and translated by the IVAP into the VH-address "12345678".

A.5.3 Service selection using ISDN supplementary services

In these configurations, ISDN supplementary services are used to convey service selection information before establishing a B-channel. This information can be used by the IVAP to select the appropriate application.

The usage of these mechanisms is independent of selection mechanisms which use B-channel protocols. Both mechanisms may be used together.

For more details on supplementary services, refer to Annex B (informative).

A.5.4 Distributed Videotex Application

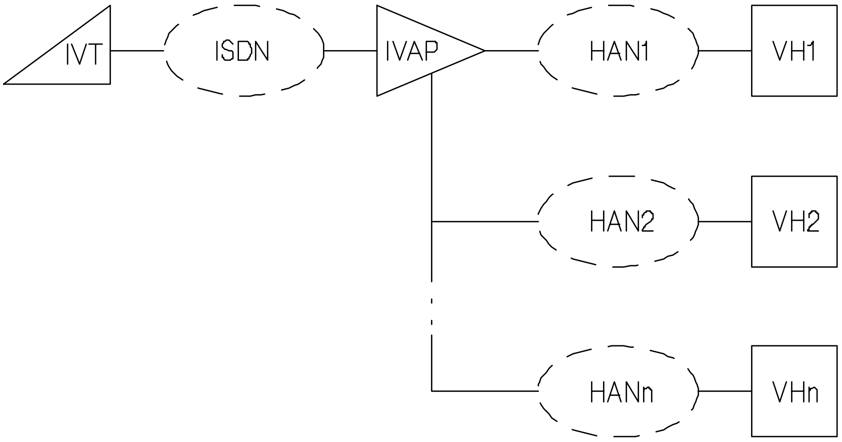


Figure A.10: Distributed Videotex Host configuration

NOTE: If one of the hosts is accessible only directly via a PSTN or PSPDN, the resulting configuration is outside the scope of this ETS.

A.5.4.1 Usage of a single VC between IVT and IVAP

The terminal is connected to the IVAP or VH1 using a VC on the B-channel. When the application requires the terminal to be connected to VH2, the IVAP should establish a VC to VH2 upon the reception of a specific request from VH1.

The terminal is linked to the IVAP via one single VC on one single B-channel, while the IVAP is connected to more than one VH. The IVAP should perform the multiplexing of informations received from VH1 and VH2 onto the one single VC to the terminal.

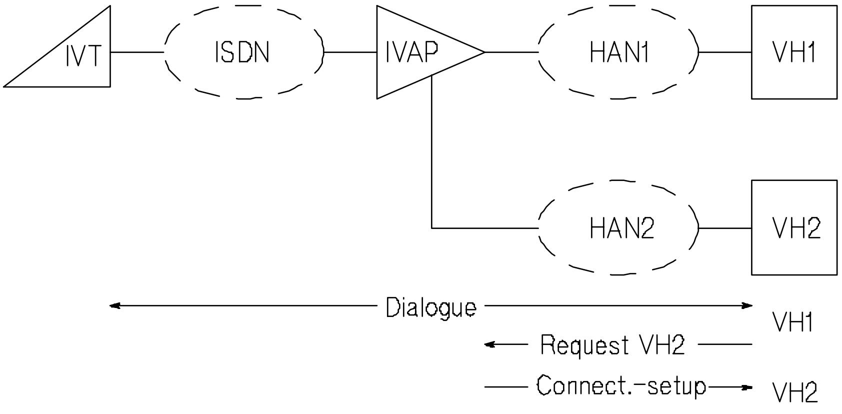


Figure A.11: Distributed hosts via a single VC

From a protocol standpoint at the terminal side, this configuration is exactly identical with the one described in Clause A.4.

A.5.4.2 Establishment of the second VC by the IVT

The terminal is connected to the IVAP or VH1 using a VC on a B-channel. After having received a specific command from the IVAP or VH1, the terminal should establish a second VC on the same B-channel to VH2.

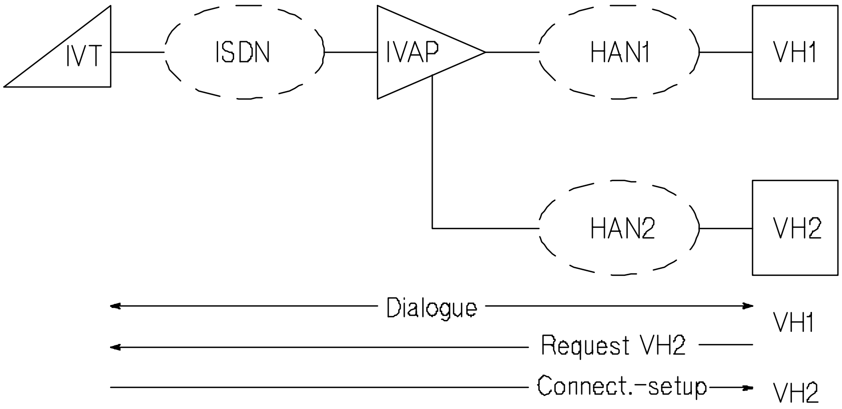


Figure A.12: Distributed hosts via two VCs initiated by the VT

The terminal should present all the informations which are received on the two different VCs. It should also be able to interpret a specific command requesting the setting up of a second VC.

A.5.4.3 Establishment of the second VC by the IVAP

The terminal is connected to the IVAP or VH1 using a VC on the B-channel. When the application requires the terminal to be connected to VH2, the IVAP establishes a second VC connection to both VH2 and the terminal on the same B-channel. If the application is run on VH1, the second VC is established after having received a specific request from VH1.

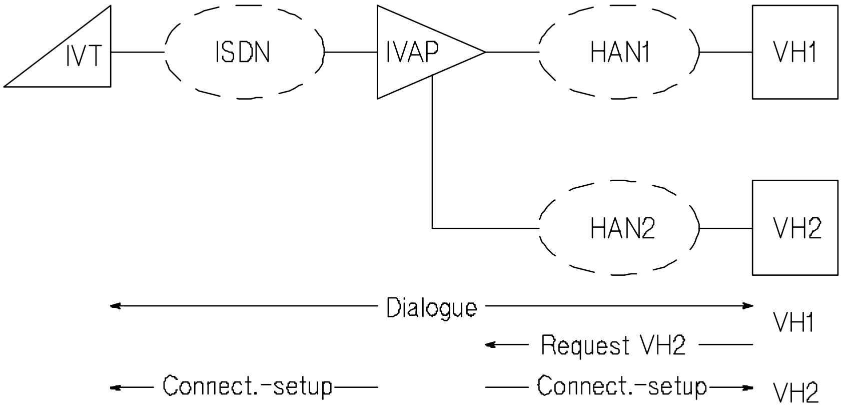


Figure A.13: Distributed hosts via two VCs initiated by the IVAP

In this configuration, the terminal should present the informations which are received on two different VCs. It shall also be able to accept a VC setup.

A.6 Host to terminal call establishment

In all configurations described above, the initiative to establish the connection may also be taken by an IVAP. Therefore, a terminal should be able to respond to incoming calls. The topology of the network behind the IVAP is irrelevant.

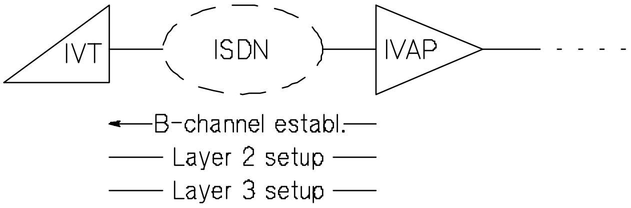


Figure A.14: Host to terminal connection establishment

After establishment of layer 3, the same procedures apply as for all above configurations, except for the connection establishment in conjunction with a direct service selection and the service selection using ISDN supplementary services.

A.7 Terminal to terminal communication

If the IVAP in the above figure A.14 is replaced by another terminal, the resulting configuration allows for a direct terminal to terminal communication, as shown in figure A.15.

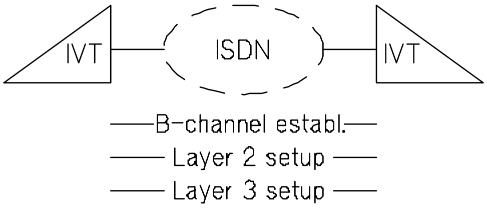


Figure A.15: Terminal to terminal communication

After establishment of layer 3, none of the service selection procedures are applicable. The Application layer protocol does not contain any special support for this configuration.

Annex B (informative): Usage of supplementary services

This annex gives an overview of the supplementary services which may be used in the ISDN syntax-based Videotex. The utilization of any supplementary service is optional.

NOTE: The use of supplementary services applies only to communications in the circuit-switched mode using a DTE/DTE connection in the Network Layer. For all other cases, the use of supplementary services is under study.

B.1 User-to-User Signalling (UUS)

The UUS supplementary service allows an ISDN user to send/receive a limited amount of information to/from another ISDN user over the D-channel. This information is passed transparently through the network. The message size is restricted to 128 octets.

The user can transfer user-to-user information (UUI) in different phases of the call depending on the service(s) to which the user subscribes. For the ISDN syntax-based Videotex, only Service 1 shall be applicable. Service 1 provides for the transfer of UUI during the setup and clearing phases of a call with UUI embedded within call control messages.

A calling party may be subscribed to UUS. The terminal should be able to enter a UUI in the SETUP message. The specification how to handle the UUS information is outside the scope of this ETS.

B.2 Supplementary services for addressing

B.2.1 Multiple Subscriber Number (MSN)

MSN allows for more than one subscriber number being allocated to a single basic access interface. By inspecting the called number on the D-channel, terminals can decide whether they should accept a call. The interpretation of the number is up to the terminals.

A calling party may be subscribed to MSN. This does not have any impact on the terminal-network interface unless the user is subscribed to CLIP. In this case, a terminal has to insert its subscriber number (one of the numbers assigned to the network interface) in the messages conveyed to the network.

B.2.2 Direct Dialling In (DDI)

DDI is intended to subaddress a specific terminal (or interface) on an integrated services private branch exchange (ISPBX), which is connected to the ISDN via a primary rate interface.

A calling party may be subscribed to DDI. This does not have any impact on the terminal-network interface unless the user is subscribed to CLIP. In this case, a terminal has to insert its subscriber number (its local extension) in the messages conveyed to the network.

B.2.3 Subaddressing (SUB)

Subaddressing (SUB) is a mechanism to convey a number of digits to the called party which may use this information for terminal selection. The maximum size of the subaddress is 20 octets. However, for a certain period of time, the size of the subaddress can be limited to a maximum of 4 octets either within certain networks or between networks.

A calling party may be subscribed to SUB. The terminal should be able to enter a subaddress in the SETUP message.

B.2.4 Call Deflection (CD)

A called party may be subscribed to CD. It can use the information conveyed through the UUS 1 supplementary service messages to deflect a call to another address. The specification how to handle the UUS information is outside scope of this ETS.

B.3 Supplementary services for identification

B.3.1 Calling Line Identification Presentation (CLIP)

A calling party may be subscribed to CLIP. If the calling party is subscribed to CLIP, the calling line number is sent to the called party as one component of the SETUP message. Generally, the calling line identification information should be inserted by the network. In specific cases, however, terminals can insert the calling line identification information in the SETUP message. This feature is mainly used on network interfaces subscribed to MSN or DDI.

B.3.2 Calling Line Identification Restriction (CLIR)

A calling party may be subscribed to CLIR. In case the user is subscribed to permanent mode, no calling line identification is transmitted to the called party. If the user is subscribed to temporary mode, he can indicate to the network, whether the calling line identification should be transmitted to the called party or not by inserting a presentation indicator in the SETUP message.

Annex C (normative): Terminal Function basic state

The initial state of a Terminal Function after the successful establishment of a virtual circuit, prior to the exchange of any further protocol elements, is called basic state. In this basic state, parameters and switches shall have the default values as defined below.

C.1 CCITT Recommendation X.3 [3] parameter default values

The default values for all CCITT Recommendation X.3 [3] parameters which are used by the ISDN syntax-based Videotex are given in table C.1.

Table C.1: CCITT Recommendation X.3 [3] ISDN syntax-based Videotex profile

Parameter reference number	Parameter description	Default value
2	Echo (M)	0
3	Selection of data forwarding character(s) (M)	0
4	Selection of idle timer delay (M)	1
11	Binary speed (M)	18
23	Size of input field (O)	0
24	End of frame signals (O)	32
25	Extended data forwarding signals (O)	0

Abbreviations:

M	Mandatory
O	Optional

C.2 SBV_DFK Default List

The SBV_DFK default list shall be empty.

C.3 SBV_TPD default state

The SBV_TPD regime shall not be established.

C.4 SBV_Escape default state

When implemented, any process related to an SBV_Escape subservice shall be inactive or shall be in its idle state.

Annex D (informative): The SBV_Escape service

D.1 Service description

D.1.1 Bulk Update

In the sense of CCITT Recommendation F.300 [1] "a Videotex service facility is an Application layer implementation in a Videotex service, providing a specific, clearly defined facility to Videotex users" (see §2.2.1).

One of the several possible service facilities is named "Videotex transaction (...) which allows users to create and/or modify information stored in a data base" (see §2.2.6). Another facility is named "Videotex data processing (...) which allows the user to employ processing and storage capacity at the host computer" (see §2.2.6).

The information elements to be handled by these service facilities may be a Videotex frame (cf. §2.4.13) for the first case or a Videotex System field (cf. §2.4.16) for the latter case.

The SBV_Escape subservices Bulk Update as defined in this clause provide for protocol elements to use these service facilities/applications for both ISDN and PSTN access in a given Videotex Service.

The Bulk Update subservices are based on a master/slave handshake procedure defined for a PSTN access which allows for the transmission of 8 bit coded data blocks under the control of the Videotex Application. As a consequence, this subservice has to follow some rules which are not conclusive for a sole ISDN access.

D.2 Procedures

D.2.1 Bulk Update

Both Terminal Function and Access Function may use the Bulk Update subservices according to the following rules.

NOTE: Additional rules concerning possible sequences of Bulk Update protocol elements are under the responsibility of the service facility provider (cf. CCITT Recommendation F.300 [1], §2.1.3).

A data block can be transmitted in portions using the BULK_DATA subservice. The portion size depends on the Videotex Service facility attached.

The last data portion is conveyed using either one of the subservices BULK_LAST or BULK_EOT. This data portion may be the only one or it may even be empty.

BULK_LAST indicates the last portion of the data block to be transmitted.

BULK_EOT indicates in addition that no further data blocks will follow.

BULK_CAN may be used to abort the transaction at any given time.

D.3 Coding

The SBV_Escape subservices are identified by values of the Service-ID parameter as defined in table D.1:

Table D.1: Values for the Service-ID parameter

SBV_Escape Subservice	Service-ID
BULK_DATA	8/0
BULK_LAST	8/1
BULK_EOT	8/2
BULK_CAN	8/3
NOTE: CCITT SG 15/VIII should be in charge of allocating code values for the SBV_Escape subservices.	

Annex E (informative): Extended data forwarding signals

The CCITT Recommendation X.3 [3] extended parameter reference 25, which is introduced in this ETS, defines a number of additional forwarding conditions.

These extended data forwarding signals are not based upon an ETS, but originate from national and de facto standards. The following overview gives some background to the introduction of these conditions:

NOTE: Information providers should only use forwarding conditions which are related to the system/profile being used.

Sequence of two characters starting with character 1/3
MINITEL terminals used in the TELETEL and other profile 2 compatible networks issue these sequences when function keys are pressed.

Character 1/10
This character (Data Collection Terminator, DCT) is used in several profile 1 orientated services to indicate to the access function, that the filling of the current form is terminated.

Character 1/12
This character (Terminator, TER) is used in several profile 1 orientated services to indicate to the access function, that the filling of the current field is terminated.

Character 2/3
This character ("#") is used in several PRESTEL (profile 3) orientated services to indicate to the access function, that the filling of the current field is terminated.

Character 5/15
This character ("_", but on various PRESTEL terminals displayed as "#") is used in several PRESTEL (profile 3) orientated services to indicate to the access function, that the filling of the current field is terminated.

Sequence of the two characters ESC "J" (1/11 4/10)
This sequence (The END BOX sequence in a 7-bit coding structure) is used in several PRESTEL (profile 3) orientated services to indicate to the access function, that the filling of the current form is terminated.

Sequence of three characters starting with the two-character sequence 2/10 3/0
These sequences are used in several PRESTEL (profile 3) orientated services to invoke special service functions (redisplay a frame, retrieve updated information, advice of charge). They do not have a specific termination character, but a fixed command length.

Annex F (informative): Bibliography

These publications serve for information. Some of them are cited at informative places in the text of this standard.

CCITT Rec. T.90 "Characteristics and protocols for terminals for telematic services in ISDN".

CCITT Rec. T.101 "International interworking for Videotex services".

ETS 300 007 "Integrated Services Digital Network (ISDN); Support of packet mode terminal equipment by an ISDN".

Final draft prETS 300 048 "Integrated Services Digital Network (ISDN); ISDN Packet Mode Bearer Services (PMBS); ISDN Virtual Call (VC) and Permanent Virtual Circuit (PVC) bearer services provided on the B-channel of the user access: basic and primary rate".

Final draft prETS 300 049 "Integrated Services Digital Network (ISDN); ISDN Packet Mode Bearer Services (PMBS); ISDN Virtual Call (VC) and Permanent Virtual Circuit (PVC) bearer services provided by the D-channel of the user access: basic and primary rate".

ETS 300 050 "Integrated Services Digital Network (ISDN); Multiple Subscriber Number (MSN) supplementary service, Service description".

ETS 300 051 "Integrated Services Digital Network (ISDN); Multiple Subscriber Number (MSN) supplementary service, Functional capabilities and information flows".

ETS 300 052 "Integrated Services Digital Network (ISDN); Multiple Subscriber Number (MSN) supplementary service, Digital Subscriber Signalling one (DSS1) protocol".

ETS 300 059 "Integrated Services Digital Network (ISDN); Subaddressing (SUB) supplementary service, Service description".

ETS 300 060 "Integrated Services Digital Network (ISDN); Subaddressing (SUB) supplementary service, Functional capabilities and information flows".

ETS 300 061 "Integrated Services Digital Network (ISDN); Subaddressing (SUB) supplementary service, Digital Subscriber Signalling one (DSS1) protocol".

ETS 300 062 "Integrated Services Digital Network (ISDN); Direct Dialling In (DDI) supplementary service, Service description".

ETS 300 063 "Integrated Services Digital Network (ISDN); Direct Dialling In (DDI) supplementary service, Functional capabilities and information flows".

ETS 300 064 "Integrated Services Digital Network (ISDN); Direct Dialling In (DDI) supplementary service, Digital Subscriber Signalling one (DSS1) protocol)".

ETS 300 089 "Integrated Services Digital Network (ISDN); Calling Line Identification Presentation (CLIP) supplementary service, Service description".

ETS 300 090 "Integrated Services Digital Network (ISDN); Calling Line Identification Restriction (CLIR) supplementary service, Service description".

ETS 300 091 "Integrated Services Digital Network (ISDN); Calling line identification, presentation and restriction (CLIP and CLIR) supplementary service, Functional capabilities and information flows".

ETS 300 092 "Integrated Services Digital Network (ISDN); Calling Line Identification Presentation (CLIP) supplementary service, Digital Subscriber Signalling one (DSS1) protocol".

ETS 300 093 "Integrated Services Digital Network (ISDN); Calling Line Identification Restriction (CLIR) supplementary service, Digital Subscriber Signalling one (DSS1) protocol".

Final draft prETS 300 106 "Terminal Equipment (TE); International Videotex interworking between a terminal and a host (T/TE 06-21)".

T/NA1(89)06 "Integrated Services Digital Network (ISDN); User to User Signalling (UUS) supplementary service, Service description".

T/NA1(89)24 "Integrated Services Digital Network (ISDN); Call Deflection (CD) supplementary service, Service description".

T/S 22-07 "Integrated Services Digital Network (ISDN); Call Deflection (CD) supplementary service, Functional capabilities and information flows".

T/S 22-17 "Integrated Services Digital Network (ISDN); User to User Signalling (UUS) supplementary service, Functional capabilities and information flows".

T/S 46-33R "Integrated Services Digital Network (ISDN); Call Deflection (CD) supplementary service, Digital Subscriber Signalling one (DSS1) protocol".

DE/TE-01012 "Integrated Services Digital Network (ISDN); ISDN lower layer protocols for ISDN Syntax-based Videotex using packet mode (X.31 Case A and B)".

ISO/IEC 8878 "Information processing systems - Data communications - Use of X.25 to provide the OSI connection-mode network service".

Annex G (informative): ETSI/CCITT cross-reference list

Table G1: ETS/CCITT Cross-References

European Telecommunication Standard	CCITT Recommendation
ETS 300 007 Integrated Services Digital Network (ISDN); Support of packet mode terminal equipment by an ISDN	CCITT Recommendation X.31 (88) Support of packet mode terminal equipment by an ISDN
ETS 300 011 Integrated Services Digital Network (ISDN); Primary rate user-network interface; Layer 1 specification and test principles	CCITT Recommendation I.431 (88) Integrated Services Digital Network (ISDN); User-network interfaces; Primary rate user-network interface - Layer 1 specification
ETS 300 012 Integrated Services Digital Network (ISDN); Basic user-network interface; Layer 1 specification and test principles	CCITT Recommendation I.430 (88) Integrated Services Digital Network (ISDN); User-network interfaces; Basic user-network interface - Layer 1 specification
ETS 300 102-1 Integrated Services Digital Network (ISDN); User-Network interface layer 3; Specification for basic call control (T/S 46-30) [CA]	CCITT Recommendation Q.931 (88) ISDN user-network interface; Layer 3 specification for basic call control
ETS 300 125 Integrated Services Digital Network (ISDN); User-Network interface data link layer specification; Application of CCITT Rec. Q.920/I.440 and Q.921/I.441 (T/S 46-20) [CC]	CCITT Recommendation Q.921 (88) ISDN user-network interface; Data link layer specification

History

Document history	
October 1991	First Edition
January 1996	Converted into Adobe Acrobat Portable Document Format (PDF)