

© British Standards Institution. No part of this publication may be photocopied or otherwise reproduced without the prior permission in writing of BSI

Draft for Development

Open Systems Interconnection: virtual terminal protocol: basic class (ISO/DIS 9041)

Projet à développer

Interconnexion de Systèmes Ouverts: protocole de terminal virtuel: jeu initial

Vornorm

Kommunikation Offener Systeme; virtuelles Terminalprotokoll, Basisklasse

Contents

	BS page		BS page
National foreword	3	8.6 Delivery control	28
Committees responsible	Back cover	8.7 Token management	29
Draft for Development		9. Procedures of subset VT-B	41
0. Introduction	4	9.1 Association establishment	41
1. Scope and field of application	5	9.2 Agreed release	41
1.1 Scope	5	9.3 Unconditional termination	41
1.2 Field of application	5	9.4 Negotiation	41
2. References	6	9.5 Data transfer	42
3. Definitions	7	9.6 Delivery control	42
3.1 Global OSI definitions	7	9.7 Token management	42
3.2 Common application service element definitions	7	10. Procedures of subset VT-C	44
3.3 Presentation Layer service element definitions	7	10.1 Association establishment	44
3.4 Virtual terminal service definitions	7	10.2 Agreed release	44
3.5 Virtual terminal protocol definitions	8	10.3 Unconditional termination	44
4. Abbreviations	9	10.4 Negotiation	44
4.1 Virtual terminal service abbreviations	9	10.5 Data transfer	45
4.2 Access rights	9	10.6 Delivery control	48
4.3 Miscellaneous	9	10.7 Token management	48
4.4 VT protocol elements	9	11. Mapping of protocol elements	50
5. Overview	11	11.1 Mapping to CASE association control services	50
5.1 Virtual terminal service summary	11	11.2 Use of presentation services	52
5.2 Model	11	12. Protocol data unit structure	54
5.3 Service assumed from the Presentation Layer	11	12.1 General format	54
5.4 Service assumed from CASE association control	14	12.2 General definitions	57
5.5 Functions of the VT protocol	15	12.3 Conceptual data store definitions	59
5.6 Protocol subsets	16	12.4 Control signal and status definitions	63
5.7 Modes of operation	17	12.5 Device object definitions	65
5.8 Access control	17	13. Conformance	69
6. Protocol elements	18	13.1 Dynamic conformance requirements	69
6.1 APQ (VT-P-ABORT)	19	13.2 Static conformance requirements	69
6.2 ASQ (VT-ASSOCIATE-REQ)	19	13.3 Protocol implementation conformance statement (PICS)	69
6.3 ASR (VT-ASSOCIATE-RESP)	20		
6.4 AUQ (VT-U-ABORT)	21	Annexes	
6.5 DAQ (VT-ACK-RECEIPT)	21	A. State tables	71
6.6 DLQ (VT-DELIVER)	21	A.1 General	71
6.7 ENQ (VT-END-NEG-REQ)	21	A.2 Parameters, VTPM rights and variables	72
6.8 ENR (VT-END-NEG-RESP)	22	A.3 Conventions for use of state tables	73
6.9 GTQ (VT-GIVE-TOKEN)	22	A.4 Actions to be taken by the VTPM	74
6.10 NAQ (VT-NEG-ACCEPT)	22	B. Defined object identifier names	119
6.11 NDQ (VT-DATA)	23	C. Examples of ASN.1 encoding for PDUs	121
6.12 NIQ (VT-NEG-INVITE)	23	C.1 Introduction	121
6.13 NJQ (VT-NEG-REJECT)	23	C.2 Notation	121
6.14 NOQ (VT-NEG-OFFER)	24	C.3 Example 1 Association request	121
6.15 RLQ (VT-RELEASE-REQ)	24	C.4 Example 2 Connect response	121
6.16 RLR (VT-RELEASE-RESP)	24	C.5 Example 3 Release request	122
6.17 RTQ (VT-REQUEST-TOKEN)	24	C.6 Example 4 Release response	122
6.18 SNQ (VT-START-NEG-REQ)	25	C.7 Example 5 Switch profile request	122
6.19 SNR (VT-START-NEG-RESP)	25	C.8 Example 6 Switch profile response	123
6.20 SPQ (VT-SWITCH-PROFILE-REQ)	25		
6.21 SPR (VT-SWITCH-PROFILE-RESP)	25	Figures	
6.22 TMQ (VT-TERMINATE-REQ)	26	1. Relationship of this standard to other ISO standards	4
6.23 TMR (VT-TERMINATE-RESP)	26	2. Model	14
6.24 UDQ (VT-UNCONTROLLED-DATA)	26		
7. Procedures	27	Tables	
8. Procedures of subset VT-A	28	1. Virtual terminal service primitives	12
8.1 Association establishment	28	2. Presentation service primitives	14
8.2 Agreed release	28	3. CASE service primitives	15
8.3 Unconditional termination	28	4. VT protocol elements	18
8.4 Negotiation	28	5. VT-association establishment procedures	29
8.5 Data transfer	28	6. Association refusal procedures	29
		7. Agreed release procedures	30

	BS page		BS page
8. Unconditional release procedures	32	25. Specific actions	90
9. S-mode data transfer procedures	33	26. Predicates	91
10. A-mode data transfer procedures	35	27. Sector 1 states—association establishment	93
11. Data transfer for uncontrolled objects procedures	38	28. Sector 1 actions	94
12. Delivery control procedures	39	29. Sector 2 states—quiescent states & profile switch	96
13. Token management procedures	40	30. Sector 2 actions	97
14. Association establishment procedures for VT-B	41	31. Sector 3 states—negotiation initiation	93
15. Termination procedures for VT-B	42	32. Sector 3 actions	98
16. Switch profile procedure	43	33. Sector 4 states—negotiation A-mode	100
17. Termination procedures for VT-C	44	34. Sector 4 actions	102
18. Establishment of multiple interaction negotiation procedures	46	35. Sector 5 states—negotiation performance	104
19. Negotiation of parameter values procedures	47	36. Sector 5 actions	105
20. Termination of multiple interaction negotiation procedures	49	37. Sector 6 states—negotiation termination	106
21. Protocol element mapping	51	38. Sector 6 actions	109
22. Incoming events	75	39. Sector 7 states—data transfer A-mode	110
23. States	81	40. Sector 7 actions	110
24. Outgoing events	84	41. Sector 8 states—data transfer S-mode	113
		42. Sector 8 actions	113
		43. Sector 9 states—connection release	114
		40. Sector 9 actions	116

This publication is not to be regarded as a British Standard. It is being issued in the Draft for Development series of publications and is of a provisional nature because the subject matter is in the process of ongoing development within the International Standards Committee (ISO/TC 97/SC 21 Information Retrieval, Transfer and Management for Open Systems Interconnection).

It should be applied on this provisional basis, so that information and experience of its practical application may be obtained.

A review of this Draft for Development will be carried out not later than two years after its publication. Notification of the start of the review period, with a request for the submission of comments from users of this Draft for Development will be made in an announcement in the appropriate issue of *BSI News*. Observations which it is felt should receive attention before the official call for comments will be welcomed.

According to the replies received, the responsible BSI Committee will judge whether the Draft for Development can be converted into a British Standard or what other action should be taken.

National foreword

This Draft for Development has been prepared under the direction of the Office and Information Standards Committee. The primary objective of this Draft for Development is to give advance guidance to product and systems designers concerned with virtual terminals in the Application Layer of the Basic Reference Model for Open Systems Interconnection (OSI).

The text of the Draft for Development is identical with that submitted to the International Organization for Standardization (ISO) for issue as ISO/DIS 9041.

For ease of reproduction the ISO text has been reproduced unchanged.

The UK has actively participated through BSI Technical Committee IST/21 in the development of the Draft International Standard. The Technical Committee will review any agreed amendments or addenda to the ISO text and will, where appropriate, update the text of this Draft for Development via an amendment or revision. When the ISO Draft reaches an appropriate level of agreement and stability, this Draft for Development may then be converted to a British Standard.

This Draft for Development is closely related to DD 129 which defines the service provided by the protocol specified in this Draft for Development.

A brochure giving details of existing and projected British Standards publications for Open Systems Interconnection (OSI) may be obtained from the Sales Department, BSI, Linford Wood, Milton Keynes, MK14 6LE, at no charge.

Press Releases for new OSI, data communications, database, and computer graphics British Standards publications will be issued to the computing press, and details of all new publications will be recorded in full in *BSI News*.

Terminology and conventions. The text of the Draft International Standard has been approved as suitable for publication as a Draft for Development without deviation. Some terminology and certain conventions are not identical with those used in Drafts for Development; attention is drawn especially to the following.

Where the words 'International Standard' appear, referring to this Draft for Development, they should be read as 'Draft for Development'.

Cross-references

International standards	Corresponding British Standards
ISO 646-1983	BS 4730 : 1985 Specification for UK 7-bit coded character set (Appendix A/Identical)
ISO 7498-1984	BS 6568 : 1984 Description of basic reference model for open systems interconnection (Identical)
ISO/DIS 8326 revised-1984	DD 111 : 1985 Open Systems Interconnection: basic connection-oriented session service definition (ISO/DIS 8326) (Identical)
ISO/DIS 8649/2	DD 131 : 1986 Open Systems Interconnection: definition of common application service elements: association control (ISO/DIS 8649/2) (Identical)
ISO/DIS 8822	DD 101 : 1986 Open Systems Interconnection: connection-oriented presentation service definition (ISO/DIS 8822) (Identical)
ISO/DIS 8824.2	DD 103 : 1986 Open Systems Interconnection: specification of Abstract Syntax Notation One (ASN.1) (ISO/DIS 8824.2) (Identical)
ISO/DIS 8825.2	DD 104 : 1986 Open Systems Interconnection: specification of basic encoding rules for Abstract Syntax Notation One (ASN.1) (ISO/DIS 8825.2) (Identical)
ISO/DIS 9040	DD 129 : 1986 Open Systems Interconnection: virtual terminal service: basic class (ISO/DIS 9040) (Identical)

The Technical Committee has reviewed the provisions of ISO 2022 and ISO 6429, to which reference is made in the text, and has decided that they are acceptable for use in conjunction with this Draft for Development.

NOTE. *Typographical error.* In the title of 1.2, delete 'Applications' and substitute 'Application'.

0 INTRODUCTION

This International Standard is one of a set of standards produced to facilitate the interconnection of computer systems. It is related to other standards in the set as defined in the Reference Model for Open Systems Interconnection. The Reference Model subdivides the area of standardisation for interconnection into a series of layers of specification, each of manageable size.

The purpose of this International Standard is to define the manner in which two protocol machines (hereafter called Virtual Terminal Protocol Machines or VTPMs) in the Application Layer of the Reference Model communicate in order to provide the Basic Class Virtual Terminal Service defined in ISO 9040 making use of the Presentation Layer Service and of the CASE association control service within the Application Layer. The relationship between these protocols and services is shown in figure 1.

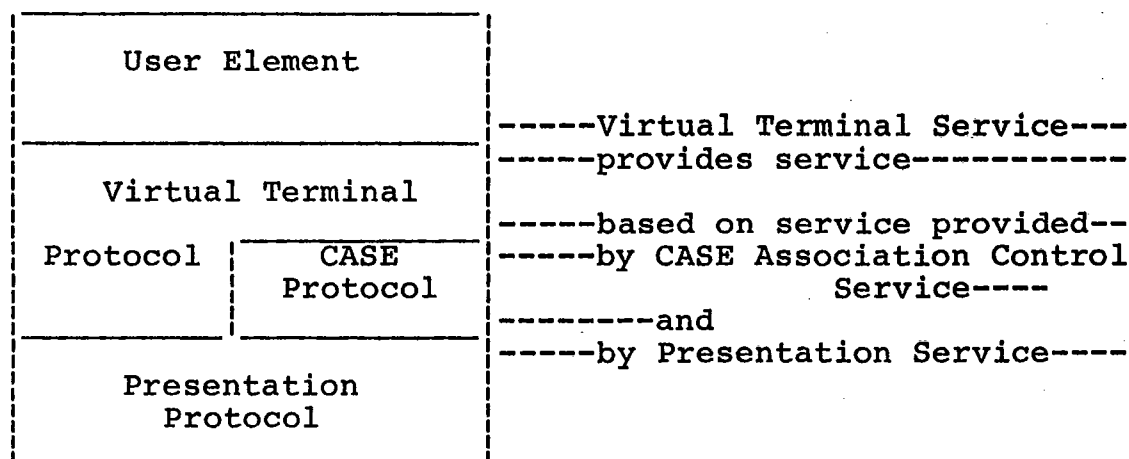


Figure 1 - Relationship of this Standard to other ISO Standards

This International Standard contains three annexes. Annex A containing the state tables and annex B containing assigned ASN.1 names are both integral parts of the standard. Annex C containing examples of encodings is not an integral part of the standard.

[ISO page 1]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2.

1 SCOPE AND FIELD OF APPLICATION

1.1 Scope

This International Standard specifies:

- a) a set of procedures for the connection-oriented transfer of data and control information between protocol machines which implement the functions of Basic Class Virtual Terminal service providers;
- b) the two modes in which those procedures operate;
- c) three subsets of that set of procedures;
- d) the means of negotiating the subset of procedures to be used by the protocol machines and the parameters of the service;
- e) the structure and mapping of protocol elements used for the transfer of data and control information.

The procedures are defined in terms of:

- f) the interactions between Virtual Terminal protocol machines through the exchange of Virtual Terminal protocol elements;
- g) the interactions between a Virtual Terminal protocol machine and the Virtual Terminal service user in the same system through the exchange of Virtual Terminal service primitives;
- h) the interactions between a Virtual Terminal protocol machine and the CASE and presentation service providers through the exchange of service primitives.

This International Standard also specifies conformance requirements (see clause 13) for systems implementing these procedures. It does not contain tests which can be used to demonstrate this conformance.

1.2 Field of Applications

These procedures are applicable to instances of communication between systems that support the Basic Class Virtual Terminal Service in the Application Layer of the Reference Model for Open Systems Interconnection and which wish to interconnect in an open systems environment.

[ISO page 2]

2 REFERENCES

- ISO 646 Information processing - ISO 7-bit Coded Character Set for Information Interchange
- ISO 2022 Information processing - ISO 7-bit and 8-bit Coded Character Sets - Code Extension Techniques
- ISO 6429 Information processing - ISO 7-bit and 8-bit Coded Character Sets - Additional Control Functions for Character Imaging I/O Devices
- ISO 7498 Information processing systems - Open systems interconnection - Basic Reference Model
- ISO 8326 Information processing systems - Open systems interconnection - Connection Oriented Session Service Definition
- DIS 8649/2 Information processing systems - Open systems interconnection - Common Application Service Elements Part 2. Association Control
- DIS 8822 Information processing systems - Open systems interconnection - Connection Oriented Presentation Service Definition
- DIS 8824 Information processing systems - Open systems interconnection - Specification of Abstract Syntax Notation One (ASN.1)
- DIS 8825 Information processing systems - Open systems interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)
- DIS 9040 Information processing systems - Open systems interconnection - Basic Class Virtual terminal Service.

[ISO page 3]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2.

3 DEFINITIONS

3.1 Global OSI Definitions

This document is based on the concepts developed in the Reference Model for Open System Interconnection, ISO 7498. It makes use of the following terms defined in that Standard:

- a) application entity;
- b) Application Layer;
- c) protocol data unit;
- d) user element.

Definitions of the following terms are given in TR 8509:

- a) primitive;
- b) confirm (primitive);
- c) indication (primitive);
- d) request (primitive);
- e) response (primitive);
- f) service provider;
- g) service user.

3.2 Common Application Service Element Definitions

Definitions of the following are to be found in ISO 8649/2:

- a) application association;
- b) application context.

3.3 Presentation Layer Service Element Definitions

Definition of the following is to be found in ISO 8822:

- a) abstract syntax.

3.4 Virtual Terminal Service Definitions

This International Standard uses the following definitions contained in the Virtual Terminal Basic Class Service Definition, (ISO 9040):

- a) repertoire;
- b) character-box graphic element;
- c) VT-association;
- d) VT-context-value;
- e) VT-environment (VTE);
- f) current-VTE;
- g) draft-VTE;
- h) full-VTE;
- i) VTE-parameter;
- j) VT-user;
- k) access-right;

[ISO page 4]

- l) token;
- m) termination-event;
- n) net-effecting;
- o) A-mode; Asynchronous Mode;
- p) S-mode; Synchronous Mode.

3.5 Virtual Terminal Protocol Definitions

For the purposes of this International Standard the following definitions apply:

3.5.1 Protocol element: an abstract unit of information, defined in clause 6, communicated between peer VTPMs, that maps

- a) directly to a CASE or presentation service primitive; and/or
- b) to a PDU type that is conveyed by a user information parameter of CASE or a user data parameter of the presentation service.

3.5.2 initiating VTPM: the VTPM that initiates an individual procedure;

3.5.3 target VTPM: the VTPM to which the protocol element initiating a procedure is directed;

3.5.4 sending VTPM the initiating VTPM for a data transfer procedure;

3.5.4 receiving VTPM the target VTPM for a data transfer procedure.

3.5.5 dynamic conformance requirements All those requirements (and options) which determine what observable behaviour is permitted in instances of communication.

3.5.6 static conformance requirements Constraints which are placed to facilitate interworking by defining the requirements for kernel sets of capabilities of an implementation.

3.5.7 protocol implementation conformance statement (PICS) A statement made by the supplier or implementor which states the capabilities and options which have been implemented, and any features which have been omitted.

[ISO page 5]

4 ABBREVIATIONS

4.1 Virtual Terminal Service Abbreviations

A-mode	Asynchronous Mode
ACS	Access Control Store
CCA	Conceptual Communication Area
CDS	Conceptual Data Store
CO	Control Object
CSS	Control, Signal and Status Store
DO	Display Object
DSD	Data Structure Definition
MIN	Multiple interaction negotiation
S-mode	Synchronous Mode
VT	Virtual Terminal
VTE	Virtual Terminal Environment
VTs	Virtual Terminal Service

4.2 Access Rights (see 5.8)

WACA	Write Access Connection (association) Acceptor
WACI	Write Access Connection (association) Initiator
WAVAR	Write Access Variable

4.3 Miscellaneous

ASN.1	See ISO 8824
CASE	See ISO 8649/2
PAB	Provider Abort
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
TWA	Two-Way Alternate
TWS	Two-Way Simultaneous
VTP	Virtual Terminal Protocol
VTPM	Virtual Terminal Protocol Machine

4.4 VT Protocol Elements

APQ	VT-P-ABORT
ASQ	VT-ASSOCIATE-REQ
ASR	VT-ASSOCIATE-RESP
AUQ	VT-U-ABORT
DAQ	VT-ACK
DLQ	VT-DELIVER
ENQ	VT-END-NEG-REQ
ENR	VT-END-NEG-RESP
GTQ	VT-GIVE-TOKEN
NAQ	VT-NEG-ACCEPT
NDQ	VT-DATA
NIQ	VT-NEG-INVITE
NJQ	VT-NEG-REJECT
NOQ	VT-NEG-OFFER
RLQ	VT-RELEASE-REQ

[ISO page 6]

RLR	VT-RELEASE-RESP
RTQ	VT-REQUEST-TOKEN
SNQ	VT-START-NEG-REQ
SNR	VT-START-NEG-RESP
SPQ	VT-SWITCH-RPROFILE-REQ
SPR	VT-SWITCH-PROFILE-RESP
TMQ	VT-TERMINATE-REQ
TMR	VT-TERMINATE-RESP
UDQ	VT-UNCONTROLLED-DATA

[ISO page 7]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2.

5 OVERVIEW

5.1 Virtual Terminal Service Summary

The protocol specified in this International Standard provides the Basic Class Virtual Terminal Service defined in ISO 9040. The VT-service primitives are listed in table 1.

5.2 Model

The Basic Class Virtual Terminal Protocol operates between two Virtual Terminal Protocol Machines (VTPMs) in the Application layer of the OSI model. Protocol elements are exchanged between these, using the services of CASE association control and of the Presentation layer as defined in ISO 8649/2 and ISO 8822.

The service is modelled as a single Conceptual Communication Area which is accessible to two communicating service users and which contains all the necessary information to allow these VT-users to derive a consistent view of the Virtual Devices which constitute the Virtual Terminal. The CCA is partitioned into four sub-areas:

- a) a conceptual data store (CDS) containing one or more display objects;
- b) a control, signalling and status store (CSS) containing a number of control objects;
- c) an access control store (ACS);
- d) a data structure definition (DSD) which parametrically defines the structure of the CDS and CSS.

The CCA is conceptually accessed by VT-users via service primitives in which information is transferred to or from the VT-users. Each VTPM includes its own Conceptual Communications Area (CCA); these are the local realizations of the CCA which is the context of the Basic Class Virtual Terminal Service (see figure 2).

5.3 Service Assumed from the Presentation Layer

The protocol specified in this International Standard assumes the use of the presentation service defined in ISO 8822. Information is transferred to and from the presentation service provider in the primitives listed in table 2.

[ISO page 8]

Table 1 - Virtual Terminal Service Primitives

Primitive		Parameters
VT-ASSOCIATE	request indication	Called Application Entity Title Calling Application Entity Title VT-class VT-class-subset VT-collision-winner VT-access-rights VT-WAVAR-initial-owner VT-profile-name VT-profile-param-offer-list
	response confirm	Responding Application Entity Title VT-class-subset VT-collision-winner VT-WAVAR-initial-owner VT-profile-param-value-list VT-result VT-user-failure-reason VT-provider-failure-reason (confirm only)
VT-RELEASE	request indication	
	response confirm	VT-result VT-user-failure-reason VT-provider-failure-reason (confirm only)
VT-U-ABORT	request indication	VT-user-failure-reason
VT-P-ABORT	indication	VT-reason
VT-SWITCH -PROFILE	request indication	VT-profile-name VT-profile-param-offer-list
	response confirm	VT-profile-param-value-list VT-result VT-user-failure-reason VT-provider-failure-reason (confirm only)

Table 1 is concluded on the next page.

[ISO page 9]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2

Table 1 (concluded)

Primitive		Parameters
VT-START-NEG	request indication	VT-profile-name VT-profile-param-offer-list
	response confirm	VT-profile-param-value-list VT-result VT-user-failure-reason VT-provider-failure-reason (confirm only)
VT-END-NEG	request indication	VT-vte-action-proposal VT-vte-action-reject-options
	response confirm	VT-vte-action-choice VT-result VT-user-failure-reason VT-provider-failure-reason (confirm only)
VT-NEG-INVITE	request indication	VT-param-ident-list
VT-NEG-OFFER	request indication	VT-param-offer-list
VT-NEG-ACCEPT	request indication	VT-param-value-list
VT-NEG-REJECT	request indication	VT-param-ident-list
VT-DATA	request indication	VT-echo-now VT-object-update VT-object-descriptor VT-object-data
VT-DELIVER	request indication	VT-ack-request
VT-ACK-RECEIPT	request indication	
VT-GIVE-TOKEN	request indication	
VT-REQUEST -TOKEN	request indication	

[ISO page 10]

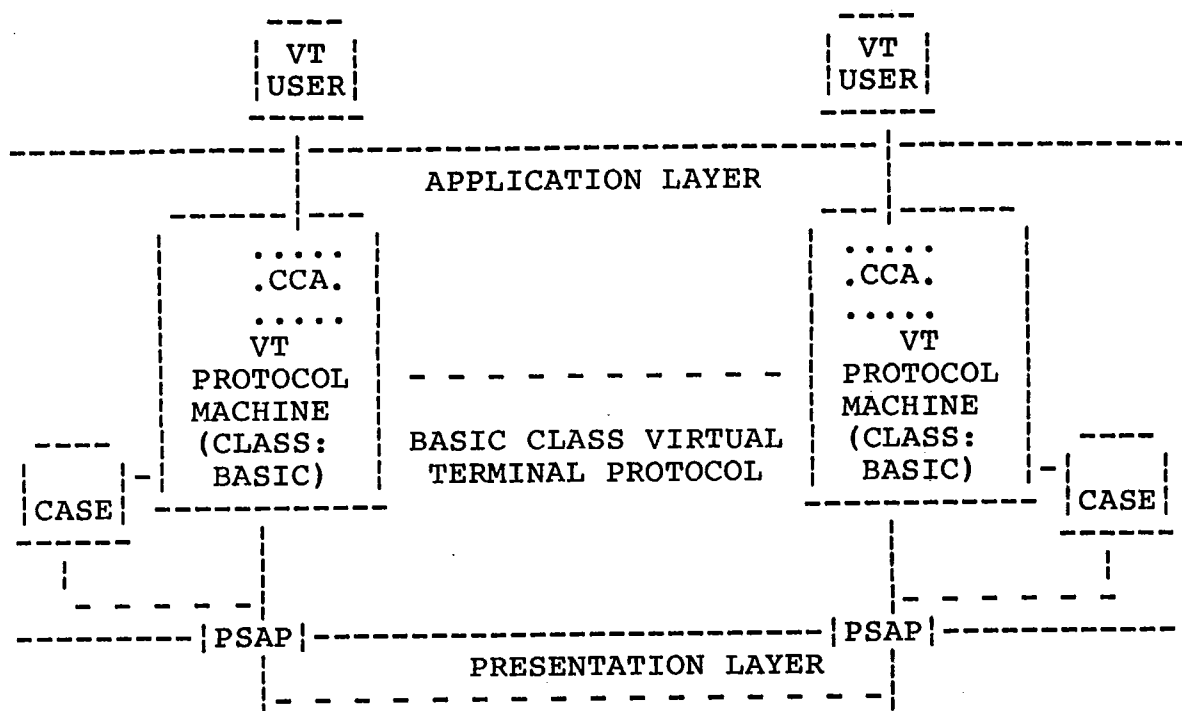


Figure 2 - Model

Table 2 - Presentation Service Primitives

P-DATA	request indication
P-TYPED-DATA	request indication
P-GIVE-TOKEN	request indication
P-TOKEN-PLEASE	request indication

5.4 Service Assumed from CASE Association Control

The protocol specified in this International Standard assumes the use of the CASE Association Control service defined in ISO 8649/2. Information is transferred to and from CASE in the service primitives listed in table 3.

[ISO page 11]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2

Table 3 - CASE Service Primitives

A-ASSOCIATE	request indication response confirm
A-RELEASE	request indication response confirm
A-ABORT	request indication
A-P-ABORT	indication

5.5 Functions of the VT Protocol

5.5.1 VT-association Establishment

The purpose of VT-association establishment is to establish a VT-association between two VT-users such that:

- a) the use of this VT-protocol is confirmed;
- b) the required dialogue discipline is agreed;
- c) a suitable transfer syntax is used;
- d) either an initial VTE is agreed, or it is agreed that the VT-environment will be negotiated later, before any data is transferred.

5.5.2 VT-association Termination

The purpose of VT-association termination is to provide for cessation of use of the VT-association.

5.5.3 Negotiation

Negotiation provides a mechanism by which the VT-users and VTPMs can agree upon the VT-environment to be used. Two forms of negotiation are supported, a single interaction profile switch and a multiple interaction dialogue.

[ISO page 12]

5.5.4 Data Transfer

The purpose of data transfer is the sending, from one VTPM to the other, of structured data representing operations upon an object or objects in the Conceptual Communications Area of the virtual terminal model.

5.5.5 Delivery Control

The purpose is:

- a) to allow the VT-user who is sending data, as above, to indicate the points in a stream of data at which the operations represented are to take effect;
- b) to allow the receipt of such a delivery point indication to be acknowledged;
- c) to allow data which updates objects with the same access rights to be grouped together and made available simultaneously to the VT-user.

5.5.6 Dialogue Control

Dialogue Control provides the VT-users with access to the lower layer functions which enforce a Two-Way Alternate discipline on the sending of some data and on the exercise of some control functions. Dialogue control is only available in the Synchronous mode of operation (see 5.7).

5.5.7 Error Handling

The purposes of error handling are detection of non-compliance with the protocol, reception of error indications from other service providers and the taking of appropriate action. The errors cause the association to be terminated with an appropriate error indication (identifying the source of error).

NOTE: The procedure for error handling is included with that for unconditional termination of an association.

5.6 Protocol Subsets

This International Standard defines three subsets of the Basic Class Virtual Terminal protocol, which correspond to the three service subsets.

[ISO page 13]

5.6.1 VT-A, providing the following functions:

- a) association establishment;
- b) association termination;
- c) data transfer;
- d) delivery control;
- e) dialogue management (S-mode only);
- f) error handling;

5.6.2 VT-B, providing:

- a) the functions of VT-A;
- b) profile switch negotiation;

5.6.3 VT-C, providing:

- a) the functions of VT-B;
- b) multiple interaction negotiation.

5.7 Modes of Operation

The protocol operates in two modes: Asynchronous and Synchronous. In the Synchronous mode the presentation connection supporting the VT communication is treated as a two-way alternate connection, and the VTPMs send data to each other alternately. In the Asynchronous mode the presentation connection is treated as a two-way simultaneous connection, and the VTPMs are not restricted in the exchange of data.

All protocol subsets apply in both modes.

5.8 Access Control

The initiation and reception of some protocol elements by a VTPM is conditional upon 'rights' held by that VTPM, conceptually in the ACS of its replica CCA. These rights are given the names:

- a) WACI;
- b) WACA;
- c) WAVAR.

They are equivalent to the access rights with the same names which the VT-service assigns to its users.

[ISO page 14]

6 PROTOCOL ELEMENTS

Table 4 lists the elements of the virtual terminal basic class protocol, defines their abbreviations and indicates to which protocol functions and subsets they are related.

This clause defines, for each element, its purpose and a list of its parameters.

Procedures specifying the use of the elements are specified in clauses 7 through 10 and their mapping to the presentation service or CASE and their ASN.1 structure are specified in clauses 11 and 12.

Table 4 - VT Protocol Elements

	abbrev.	full name	Cross ref.	Subset
Association establishment	ASQ	VT-ASSOCIATE-REQ	6.2	
	ASR	VT-ASSOCIATE-RESP	6.3	
Association termination	RLQ	VT-RELEASE-REQ	6.15	Subsets VT-A, VT-B and VT-C
	RLR	VT-RELEASE-RESP	6.16	
	TMQ	VT-TERMINATE-REQ	6.22	
	TMR	VT-TERMINATE-RESP	6.23	
	AUQ	VT-U-ABORT	6.4	
	APQ	VT-P-ABORT	6.1	
Data transfer	NDQ	VT-DATA	6.11	Subsets VT-B and VT-C
	UDQ	VT-UNCONTROLLED-DATA	6.24	
Data delivery control	DLQ	VT-DELIVER	6.6	
	DAQ	VT-ACK-RECEIPT	6.5	
Dialogue management	GTQ	VT-GIVE-TOKEN	6.9	
	RTQ	VT-REQUEST-TOKEN	6.17	
Negotiation	SPQ	VT-SWITCH-PROFILE-REQ	6.20	Subsets VT-B and VT-C
	SPR	VT-SWITCH-PROFILE-RESP	6.21	
	SNQ	VT-START-NEG-REQ	6.18	Subset VT-C only
	SNR	VT-START-NEG-RESP	6.19	
	ENQ	VT-END-NEG-REQ	6.7	
	ENR	VT-END-NEG-RESP	6.8	
	NIQ	VT-NEG-INVITE	6.12	
	NOQ	VT-NEG-OFFER	6.14	
	NAQ	VT-NEG-ACCEPT	6.10	
	NJQ	VT-NEG-REJECT	6.13	

[ISO page 15]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2.

6.1 APQ (VT-P-ABORT)

6.1.1 Purpose:

To force unconditional termination of the VT-association (initiated by the VTPM).

6.1.2 Parameters:

- reason: takes one of the values "local error" or "protocol error".

NOTE: The failure of some lower layer or some other service of the OSI model, which results in the loss of the VT-association, will be signalled to both VTPMs. Since such a signal is not passed between VTPMs it is not identified as a protocol element; it is however, a significant event and is identified by the name Provider Abort (PAB).

6.2 ASQ (VT-ASSOCIATE-REQ)

6.2.1 Purpose

To request establishment of a VT-association.

6.2.2 Parameters

- called application entity title: specifies the application entity with which a VT-association is to be established;
- calling application entity title: identifies the application entity initiating the VT-association;
- class: specifies the VT service class to be provided; takes the value "Basic";
- class-subset: specifies the subset of the Basic Class VTS proposed for use (see 5.6); takes one of the values "A", "B" or "C";
- collision-winner: in A-mode, the VTPM whose request is honoured in the event of a collision; takes one of the values "initiator", "acceptor" or "acceptor choice";
- access-rights: specifies the access rights which will exist for the duration of the VT-association; takes one of the values WAVAR or WACI+WACA;
- WAVAR-initial-owner: defines the VTPM that will initially own the WAVAR token if this exists; takes one of the values "initiator", "acceptor", or "acceptor choice";

[ISO page 16]

- profile-name: the name of the profile (see ISO 9040); if value is absent, the default profile defined in ISO 9040 for A-mode or for S-mode, as selected by the access rights parameter, is used;
- profile-param-offer-list: a list of items in which each profile parameter under negotiation is either a value, list of values or range of values;
- facility-set: specifies the selected facility-set of the current VT service class; takes the value "initial".
- protocol-version: takes the value iso-BCVTP for this version of the standard.

6.3 ASR (VT-ASSOCIATE-RESP)

6.3.1 Purpose:

To complete or refuse establishment of a VT-association.

6.3.2 Parameters:

- responding application entity title: identifies the application entity which is responding to a VT-association request;
- class-subset: As in 6.2 but not present if the value of result is "failure";
- collision-winner: As in 6.2 except that the value "acceptor choice" is not allowed; not present if the value of result is "failure";
- WAVAR-initial-owner: As in 6.2 except that the value "acceptor choice" is not allowed;
- profile-param-value-list: a list of parameters and unique values of each parameter selected from the list or range of values in the profile parameter offer list, not present if the value of result is "failure";
- result: takes one of the values "success", "success with warning" or "failure";
- failure-reason: conveys either a text string supplied by the responding VT-user or the value "VTE-incomplete";
- protocol-version: As in 6.2.

[ISO page 17]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2.

6.4 AUQ (VT-U-ABORT)

6.4.1 Purpose:

To force unconditional termination of the VT-association (initiated by the VT-user).

6.4.2 Parameters:

- VT-user-failure-reason: user provided text string.

6.5 DAQ (VT-ACK-RECEIPT)

6.5.1 Purpose:

To Acknowledge a delivery point identified by the receipt of a DLQ.

6.5.2 Parameters:

None.

6.6 DLQ (VT-DELIVER)

6.6.1 Purpose:

To designate delivery points in the stream of NDQ PDUs and, if required, to request acknowledgement of receipt.

6.6.2 Parameters:

- ack-request: a single boolean indicating whether or not an acknowledgement is explicitly required.

6.7 ENQ (VT-END-NEG-REQ)

6.7.1 Purpose:

To request the synchronized termination of a multiple interaction negotiation and to cause a transition to the data handling phase.

6.7.2 Parameters:

- vte-action-proposal: if present takes the values "switch" or "restore". Absence implies responding user has the choice of value;

[ISO page 18]

- vte-action-reject-options: if present, takes the value "retain-draft-vte" or "discard-draft-vte" with absence implying that the responding user has the choice of actions.

6.8 ENR (VT-END-NEG-RESP)

6.8.1 Purpose:

To respond to a request to terminate negotiation.

6.8.2 Parameters:

- vte-action-choice: may be present if the value of result is "success" and then takes the values "switched" or "restored".
- result: may take the values "success", "success-with-warning" or "failure" ;
- failure-reason: conveys either a text string supplied by the responding VT-user or the value "VTE incomplete";

6.9 GTQ (VT-GIVE-TOKEN)

6.9.1 Purpose:

To pass the token to the peer VTPM.

6.9.2 Parameters:

None.

6.10 NAQ (VT-NEG-ACCEPT)

6.10.1 Purpose:

To select values for one or more of the VTE-parameters specified in previous VT-NEG-OFFER protocol elements.

6.10.2 Parameters:

- param-value-list: a list of VTE-parameters and corresponding values for those parameters selected from the alternatives specified previous VT-NEG-OFFER.

[ISO page 19]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2.

6.11 NDQ (VT-DATA)

6.11.1 Purpose:

To update objects in a "controlled" manner and, for A-mode, to indicate if echoing may take place following processing of a data item.

6.11.2 Parameters:

A sequence of one or more data items each of which itself is a sequence of one or more operations each acting upon a display object (DO) in the Conceptual Data Store or upon a Control Object in the CSS. Each operation is expressed as an identifier together with parameter values when required for that operation. Each data item is marked with an "echo-now" boolean.

The operations are as defined for display objects and control objects in Basic Class VT Service (ISO 9040).

6.12 NIQ (VT-NEG-INVITE)

6.12.1 Purpose:

To request the peer to provide values for specified VTE-parameters, once an agreement to enter into multiple interaction negotiation has been reached.

6.12.2 Parameters:

- param-ident-list: the names of VTE-parameters for which values are to be provided.

6.13 NJQ (VT-NEG-REJECT)

6.13.1 Purpose:

To reject VTE-parameters contained in previous NOQs whose values and/or value ranges are unacceptable.

6.13.2 Parameters:

- param-ident-list: the VTE-parameters whose values are being rejected.

[ISO page 20]

6.14 NOQ (VT-NEG-OFFER)

6.14.1 Purpose:

To pass a list of VTE-parameters and parameter values, lists, or ranges to the peer once an agreement to enter into multiple interaction negotiation has been reached.

6.14.2 Parameters:

- param-offer-list: a list of VTE-parameters and corresponding values, lists or ranges for those parameters.

6.15 RLQ (VT-RELEASE-REQ)

6.15.1 Purpose:

To request the orderly termination of a VT-association.

6.15.2 Parameters:

none.

6.16 RLR (VT-RELEASE-RESP)

6.16.1 Purpose:

To accept or reject a request for orderly termination of a VT-association.

6.16.2 Parameters:

- result: takes the values "success" or "failure" only;
- failure-reason: conveys either a text string provided by the responding VT-user or the value "collision-detected".

6.17 RTQ (VT-REQUEST-TOKEN)

6.17.1 Purpose:

To request possession of the protocol token.

6.17.2 Parameters:

none.

[ISO page 21]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2.

6.18 SNQ (VT-START-NEG-REQ)

6.18.1 Purpose:

To request the establishment of the Negotiation Active phase.

6.18.2 Parameters:

- profile-name: the name of the profile (see ISO 9040). If value is absent, the current-VTE serves as the basis for negotiation;
- profile-param-offer-list: as in 6.2.

6.19 SNR (VT-START-NEG-RESP)

6.19.1 Purpose:

To indicate success or failure of the attempt to establish the Negotiation Active phase.

6.19.2 Parameters:

- profile-param-value-list: as in 6.3;
- result: takes the values "success", or "failure";
- failure-reason: conveys either a text string supplied by the responding VT-user or one of the values "collision-detected" or "profile-not-supported".

6.20 SPQ (VT-SWITCH-PROFILE-REQ)

6.20.1 Purpose:

To request a switch to a new profile for the Full-VTE.

6.20.2 Parameters:

- profile-name: As in 6.2;
- profile-param-offer-list: as in 6.2.

6.21 SPR (VT-SWITCH-PROFILE-RESP)

6.21.1 Purpose:

To indicate success or failure of the negotiation attempt.

6.21.2 Parameters:

- profile-param-value-list: as in 6.3;
- result: takes the values "success" or "failure";
- failure-reason: conveys either a text string supplied by the responding VT-user or one of the values "collision-detected", "parameter-not-supported", "parameter-combination-not-supported" or "profile-not-supported".

6.22 TMQ (VT-TERMINATE-REQ)

6.22.1 Purpose:

To continue orderly termination of a VT-association after acceptance of a request for such termination.

6.22.2 Parameters:

none

6.23 TMR (VT-TERMINATE-RESP)

6.23.1 Purpose:

To complete orderly termination of a VT-association.

6.23.2 Parameters:

none

6.24 UDQ (VT-UNCONTROLLED-DATA)

6.24.1 Purpose:

To convey updates to Control Objects for which the Access-control parameter has the value "uncontrolled".

6.24.2 Parameters:

A sequence of (object id., operation) pairs.

7 PROCEDURES

Clauses 8 to 10 specify the procedures of the Basic Class VTP as sequences of actions performed by the peer VTPMs. The sequences are presented in tabular form, indicating:

- a) step: numerical order in the sequence; when a sequence branches, the first step in each branch is suffixed with a different letter and subsequent steps in each branch retain this suffix (for example, if a branch occurs after step 4, the next steps are numbered 5a and 5b; 6a then follows 5a and 6b follows 5b);
- b) VTPM: distinguishes the VTPM by its role in the procedure concerned (e.g. initiating VTPM, target VTPM) or by reference, in S-mode, to the holder of the token;
- c) Event: the receipt of a service primitive or protocol data unit;
- d) Action: specifies the action to be taken by the VTPM in response to the event.

Conditions causing branches are included in the description of the event. Where an action concludes the sequence for an individual branch but is not the last one tabulated, the phrase "sequence ends" is appended. Information that is neither an event nor an action is placed within parentheses.

8 PROCEDURES OF SUBSET VT-A

This clause specifies the procedures of subset VT-A in terms of the sequences of actions to be performed by the communicating VTPMs.

8.1 Association Establishment

Establishment of a VT-association as required by a VT-ASSOCIATE request primitive is effected by the procedures of table 5. On successful completion, any of the available facilities (i.e., data transfer, delivery control, token management and termination) may be used.

Table 6 specifies the procedures for association refusal in terms of variant actions in place of those of table 5.

8.2 Agreed Release

The procedures of table 7 are used to effect a release of the VT-association. In S-mode a VTPM initiating a release action must hold the token. In A-mode collisions may occur between an attempted release and another attempted release or between an attempted release and a delivery action requiring confirmation. The refusal and collision portions of the table are each variations applied to the first portion.

8.3 Unconditional Termination

Unconditional termination of a VT-association may be initiated by a VT-user, a VTPM or by Another Service Provider. The procedures are given in table 8.

8.4 Negotiation

Subset VT-A does not include Negotiation.

8.5 Data Transfer

The procedures for the transfer of data are defined in tables 9, 10 and 11. The procedures differ according to the type of data and the direction of transfer relative to the position of the token.

8.6 Delivery Control

The procedures defined in table 12 apply if Simple Delivery Control or Quarantine Delivery Control have been agreed upon. The acknowledgement procedures only apply if specifically requested in the VT-DELIVER primitive.

[ISO page 25]

Other service elements cause the forwarding and delivery actions to occur but do not cause a VT-DELIVER indication to be issued. The tables appropriate to these elements specify when this is to occur.

8.7 Token Management

The token management procedures allow a VTPM to pass the token to the peer VTPM or to request the token from the peer VTPM. They apply to S-mode only. These procedures may be found in table 13.

Table 5 VT-Association Establishment Procedures

Step	VTPM	Event	Action
1	initiating	VT-ASSOCIATE request acceptable to initiating VTPM	Send ASQ.
2a	target	ASQ acceptable to target VTPM	Issue VT-ASSOCIATE indication.
3aa	target	VT-ASSOCIATE response with result="success"	Send ASR with result="success".
4aaa	initiating	ASR with result="success" and ASR contents acceptable to initiating VTPM	Issue VT-ASSOCIATE confirm with result="success".

Table 6 Association Refusal Procedures

Step	VTPM	Event	Action
2b	target	ASQ not acceptable to target VTPM	Send ASR with result="failure".
3ab	target	VT-ASSOCIATE response result="failure"	Send ASR with the same result value.
4ab/ 3b	initiating	ASR with result="failure"	Issue VT-ASSOCIATE confirm with the same result value. Sequence ends.

[ISO page 26]

Table 6 (Concluded)

Step	VTPM	Event	Action
4aab	initiating	ASR with result="success" and ASR contents not acceptable to initiating VTPM	Issue VT-ASSOCIATE confirm with result="failure". send APQ. Sequence ends.

Table 7 Agreed Release Procedures

Step	VTPM	Event	Action
Agreed Release			
1	initiating	VT-RELEASE request	Forward data as specified in Table 9 or 10. Send RLQ.
2a	target	RLQ without collision	Perform data delivery as specified in Table 9 or 10. Issue a VT-RELEASE indication.
3aa	target	VT-RELEASE response result="success"	Forward data as specified in Table 9 or 10. Send RLR with result="accept".
4aaa	initiating	RLR with result="accept"	Deliver data as specified in Table 9 or 10. Send TMQ.
5aaa	target	TMQ	Send TMR.
6aaa	initiating	TMR	Issue VT-RELEASE confirm with result="success". Sequence ends.
Refusal of Release			
3ab	target	VT-RELEASE response result="failure"	Send an RLR with result="reject".

[ISO page 27]

Table 7 (Concluded)

Step	VTPM	Event	Action
4ab	initiating	RLR result="reject"	Issue a VT-RELEASE confirm with result="failure". (Take no action in respect of data in transit).
Release Collision			
2b	target	RLQ when delivery acknowledgement is pending	none.
3b	initiating	DLQ with request for acknowledgement instead of RLR	Issue VT-RELEASE confirm primitive with result="failure" and reason="collision detected". Perform normal delivery action. Sequence ends.
4aab	initiating	RLQ instead of RLR and this VTPM is not the designated the collision winner	Issue VT-RELEASE confirm with result="failure" and reason="collision detected". Proceed with step 2a of of table 7
4aac	initiating	RLQ instead of RLR and this VTPM is designated the collision winner	None (continue to await RLR).
4aad	initiating	DLQ with request for acknowledgement instead of RLR	Issue VT-RELEASE confirm with result="failure" and reason="collision detected". Proceed with step 2 of of table 12

[ISO page 28]

Table 8 Unconditional Release Procedures

Step	VTPM	Event	Action
User Initiated Termination			
1	initiating	VT-U-ABORT request	Send AUQ with reason taken from the VT-U-ABORT request (Any data in transit or other actions in progress are abandoned).
2	target	AUQ and VT-ASSOCIATE indication has been issued by the VTPM and VT-User has issued neither a VT-U-ABORT request nor a VT-RELEASE response with result="success"	Issue a VT-U-ABORT indication with reason parameter from AUQ.
VTPM Initiated Termination			
1	initiating	Some internal problem which prevents the VTPM from continuing normally	Send APQ with reason="local-error". Issue a VT-P-ABORT indication with appropriate reason parameter (Any data in transit or other actions in progress are abandoned).
2	target	APQ with reason="local-error" and VT-ASSOCIATE indication has been issued by the VTPM and VT-User has issued neither a VT-U-ABORT request nor a VT-RELEASE response with result="success"	Issue a VT-P-ABORT indication with appropriate reason parameter.

[ISO page 29]

Table 8 (Concluded)

Step	VTPM	Event	Action
Supporting Service Initiated Termination			
1	EITHER	PAB	Abandon any data in transit and actions in progress. Issue VT-P-ABORT indication with reason information as provided by the supporting service.

Table 9 S-mode Data transfer Procedures

Step	VTPM	Event	Action
1	sending	VT-DATA request addressing a "controlled" object	Store the specified updates in the CCA objects. If any of the objects updated have the "trigger" characteristic Send any NDQ PDUs which have not yet been transmitted. Send GTQ.
2	sending	At the convenience of the VTPM	Send NDQ containing the updates in a manner so as to maintain the relative sequence of CCA object updates.
3a	sending	VT-DELIVER, VT-GIVE-TOKEN or VT-RELEASE request	Send any NDQs not yet forwarded. Take action appropriate to request primitive.
3b	sending	VT-DATA request addressing object with "trigger" characteristic	Send any NDQs not yet forwarded. Send GTQ.

[ISO page 30]

Table 9 (Continued)

Step	VTPM	Event	Action
4aa/ 4ba	receiving	NDQ when no-delivery-control is in effect	Apply the updates to the relevant CCA object so as to maintain the order of updates (these updates are applied at the convenience of the VTPM). Issue VT-DATA indication maintaining the 1:1 relation of requests and indications.
4ab/ 4bb	receiving	NDQ when simple-delivery-control is in effect	Apply the updates to the CCA so as to maintain their order. (these updates are applied at the convenience of the VTPM).
5ab/ 5bb	receiving	At the convenience of the VTPM	Issue VT-DATA indication maintaining the 1:1 relation of requests and indications.
6ab/ 6bb	receiving	DLQ, GTQ or RLQ	Apply all outstanding updates and issue corresponding VT-DATA indications. Take action appropriate to received protocol element. Sequence ends. (If a GTQ is received as a result of updating a "trigger" control object, the 1:1 relation between requests and indications is not preserved as the single update to a "trigger" object results in both a VT-DATA indication and a VT-GIVE-TOKEN indication).

[ISO page 31]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2

Table 9 (Concluded)

Step	VTPM	Event	Action
4ac/ 4bc	receiving	NDQ when quarantine-delivery-control is in effect	Store updates but do not apply them to the CCA objects.
5ac/ 5bc	receiving	DLQ, GTQ or RLQ	Apply stored updates to the CCA. Issue VT-DATA indications maintaining the relative sequence of updates; the number of indications issued may not be 1:1 with the number of VT-DATA requests originally issued, due to possible net-effecting. Sequence ends.

Table 10 A-mode Data transfer Procedures

Step	VTPM	Event	Action
1	sending	VT-DATA request	Store the specified update in the CCA.
2	sending	At the convenience of the VTPM	Send NDQ expressing the stored updates and maintaining the relative sequence of display and control object updates. (If quarantine-delivery-control is in effect there need not be a 1:1 correspondence between VT-DATA requests and NDQs (net effecting)).

[ISO page 32]

Table 10 (Continued)

Step	VTPM	Event	Action
3a	receiving	NDQ when no-delivery-control is in effect	Apply the updates to relevant CCA objects so as to maintain the order of updates. (these updates are applied at the convenience of the VTPM). Issue VT-DATA indication maintaining the 1:1 relation of requests and indications.
3b	receiving	NDQ when simple-delivery-control is in effect	Apply the updates in the NDQ to the CCA so as to maintain the order of updates. (these updates are applied at the convenience of the VTPM). Issue VT-DATA indication maintaining the 1:1 relation of requests and indications. (The receipt of other PDUs will cause the updates to be applied and send NDQ actions as well, see DLQ and RLQ)
3c	receiving	NDQ when quarantine-delivery-control is in effect	Store updates in NDQ but do not apply to the CCA objects. (The number of indications may not be 1:1 with the number of VT-DATA requests, due to possible net effecting. For the timing of this see events for DLQ and RLQ)

[ISO page 33]

This publication is a Draft for Development and is therefore not to be regarded as a British Standard. See page 2.

Table 10 (Continued)

Step	VTPM	Event	Action
4	sending	VT-DELIVER, or VT-RELEASE request, VT-RELEASE response result="accepted", or a VT-DATA request addressing a control object with "trigger" char- acteristic	Send NDQs in accordance with any CCA updates received in preceding VT-DATA requests but not yet forwarded by the VTPM, maintain- ing the relative se- quence of display and control object updates. In the case of the "trigger" object, the update to that object is transmitted as the whole or part of a final NDQ. (For the VT-DELIVER, VT-GIVE-TOKEN and VT- RELEASE cases further action is specified in tables 7,8,12 & 13)
5	receiving	RLQ, DLQ or RLR with result= "accepted", or NDQ updating a "trigger" control object	Update the local CCA replica with any up- dates specified in pre- vious NDQs but not yet implemented maintaining the relative sequence of display and control object updates. Issue VT-DATA indica- tions in accordance with updates to CCA specified in previous NDQs not yet notified to the VT-user and res- pecting the relative sequence of updates to the CDS and CSS. (If quarantine-delive- ry-control is in ef- fect, there is not ne- cessarily a 1:1 corres- pondence between the VT-DATA indications and

[ISO page 34]