

© 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
service: basic class (ISO/DIS 9040)**

Projet à développer

Interconnexion de systèmes ouverts: service de terminal virtuel: jeu initial

Vornorm

Kommunikation Offener Systeme; virtueller Terminaldienst, Basisklasse

Contents

	BS page		BS page
National foreword	3		
Committees responsible	Back cover		
Draft for Development			
0 Introduction	4	17.3 Tertiary VTE-parameter	42
1 Scope and field of application	5	18 Operations on display objects	43
1.1 Scope	5	18.1 Addressing operations	43
1.2 Field of application	5	18.1.1 Implicit addressing	43
2 References	6	18.1.2 Explicit addressing	43
3 Definitions	7	18.1.3 Update-window mechanism	44
3.1 Global OSI definitions	7	18.2 Update operations	46
3.2 Common application service element definitions	7	18.2.1 Text operation	46
3.3 Virtual terminal service definitions	8	18.2.2 Attribute operation	46
4 Abbreviations	14	18.2.3 Erase-X-array operation	49
4.1 General	14	18.2.4 Erase-Y-array operation	50
4.2 Modes of operation	14	18.2.5 Erase-Z-array operation	51
4.3 VTE model components	14	19 Control object VTE-parameters	53
4.4 Access-rights	14	19.1 Usage and effects of control object VTE-parameters	53
5 Conventions	14	19.2 Standard control objects	55
		19.2.1 Standard control objects specific to A-mode	55
Section one. Overview		20 Device object VTE-parameters	57
6 General features	15	20.1 Use of device object "assignment" VTE-parameters	59
6.1 Introduction	15	20.2 Default control object VTE-parameters	60
6.2 Features of the service	15	20.3 Minimum length VTE-parameters	60
6.3 VTE parameters	16	20.4 Termination VTE-parameters	60
6.4 VT profiles	16	21 Delivery control, synchronisation and net-effecting	63
6.5 Dialogue control	16	21.1 No delivery-control	63
7 Communication facilities	17	21.2 Simple delivery-control	64
7.1 Establishment facility	17	21.3 Quarantine delivery-control	64
7.2 Termination facility	17	21.4 Implicit delivery	65
7.3 Negotiation facility	17		
7.4 Data transfer facility	17	Section four. Communication facilities	
7.5 Delivery control facility	18	22 Communication model	67
7.6 Access-right management facility	18	23 VT services	68
		24 VT service sequences	70
Section two. General		24.1 Phases	70
8 Modes of operation	19	24.2 Phase transitions	70
8.1 S-mode	19	24.3 Availability and usage conditions of VT services	72
8.2 A-mode	19	24.4 Service collisions in A-mode	74
9 Access-rights	20	25 Establishment facility	75
10 Subsets and negotiation	22	25.1 VT-associate service	75
10.1 Service subsets	22	26 Termination facility	81
10.2 Principles of negotiation	22	26.1 Services	81
10.2.1 Switch profile negotiation	23	26.2 VT-RELEASE service	81
10.2.2 Multiple interaction negotiation	23	26.3 VT-U-ABORT service	82
11 Service profiles	25	26.4 VT-P-ABORT service	84
		27 Negotiation facilities	85
Section three. VT environment		27.1 Switch profile negotiation	85
12 The VTE model	26	27.1.1 VT-SWITCH-PROFILE service	85
12.1 Structure	26	27.2 Multiple interaction negotiation	87
13 Display objects	28	27.2.1 VT-START-NEG service	87
13.1 Structure	28	27.2.2 VT-END-NEG service	89
13.2 Attributes	28	27.2.3 VT-NEG-INVITE service	92
14 Control objects	30	27.2.4 VT-NEG-OFFER service	94
15 Device objects	32	27.2.5 VT-NEG-ACCEPT service	95
16 Directed graph of VTE-parameters	33	27.2.6 VT-NEG-REJECT service	96
17 Display object VTE-parameters	35	27.3 Sequence control for multiple interaction negotiation	97
17.1 Primary VTE-parameters	35	27.3.1 Specific information for S-mode	98
17.2 Secondary VTE-parameters	36	27.3.2 Specific information for A-mode	98
17.2.1 Addressing VTE-parameters	37	28 Data transfer facility	100
17.2.2 Repertoire-assignment VTE-parameters	38	28.1 VT-DATA service	100
17.2.3 Emphasis and colour "assignment" VTE-parameters	41		

	BS page		BS page
29 Delivery control facility	102	B.8 Profile VT-B-R-Pr-A2 (r1, r2, r3, r4, r5)	124
29.1 VT-DELIVER service	102	B.9 Profile VT-B-R-Pr-A3 (r1, r2, r3)	127
29.2 VT-ACK-RECEIPT service	103	B.10 Profile VT-B-R-Pr-A4 (r1, r2, r3, r4)	130
30 Access-right management facility	105	B.11 Profile VT-B-R-Pr-A5 (r1) [transparent profile]	133
30.1 VT-GIVE-TOKENS service	105	B.12 Profile VT-B-R-Pr-A6 (r1)	135
30.2 VT-REQUEST-TOKENS	105	C Explanatory notes	
Annexes		C.1 Types of VT communication supported	140
A Default profiles	106	C.2 Aid to understanding the role of display objects	140
A.1 Notation for definition of profiles	106	C.3 Relation of update-window dimensions to buffering	141
A.1.1 Profile names	106	C.4 Device object semantics	141
A.1.2 Profile definitions	107	C.5 Echoing	141
A.2 S-mode default profile, VT-B-R-PR-SD	108	C.6 Echo control	142
A.3 A-mode default profile, VT-B-R-PR-AD	109	C.7 Echo control algorithm	143
B Draft basic class VT profiles for registration	111	C.8 Termination conditions	143
B.1 Introduction to profile definitions	112	C.9 Synchronisation of update delivery	144
B.2 Notation for definition of profiles	113	C.10 Multiple interaction negotiation	145
B.2.1 Profile names	113	C.11 Semantics of display objects	145
B.2.2 Profile parameter names	113	C.12 Repertoires	146
B.2.3 Profile definitions	113	C.13 Use of ISO 6429 additional controls in repertoire	146
B.3 Profile VT-B-R-Pr-S1 (r1, r2)	114	C.14 Font assignment VTE-parameters	147
B.4 Profile VT-B-R-Pr-S2 (r1, r2, r3)	116	C.15 Net-effecting	147
B.5 Profile VT-B-R-Pr-S3 (r1, r2, r3)	118		
B.6 Profile VT-B-R-Pr-S4 (r1, r2, r3)	120		
B.7 Profile VT-B-R-Pr-A1 (r1, r2, r3)	122		

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 9040.

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 130 which specifies the protocol providing the service defined 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.

Wherever 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/DTR 8509	DD 108 : 1986 Open Systems Interconnection: service conventions (ISO/DTR 8509) (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 8824.2	DD 103 : 1986 Open Systems Interconnection: specification of abstract syntax notation one (ASN.1) (ISO/DIS 8824.2) (Identical)

The Technical Committee has reviewed the provisions of ISO 2022, ISO 2375, ISO 6429 and the International Register of Coded Character Sets to be used with Escape Sequences, to which reference is made in the text, and has decided that they are acceptable for use in conjunction with this Draft for Development.

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 (OSI). 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 service provided in the Application Layer by the Basic Class Virtual Terminal (VT) Service. The Basic Class Virtual Terminal Service is provided by the Basic Class Virtual Terminal Protocol making use of services available from the Common Application Service Element (CASE) in the Application Layer and the Presentation Service. This International Standard also defines the Basic Class Virtual Terminal Service characteristics which the VT-user may exploit. The relationship between the standards for Virtual Terminal Service, Virtual Terminal Protocol, CASE and the user of the Virtual Terminal Service is illustrated in figure 1.

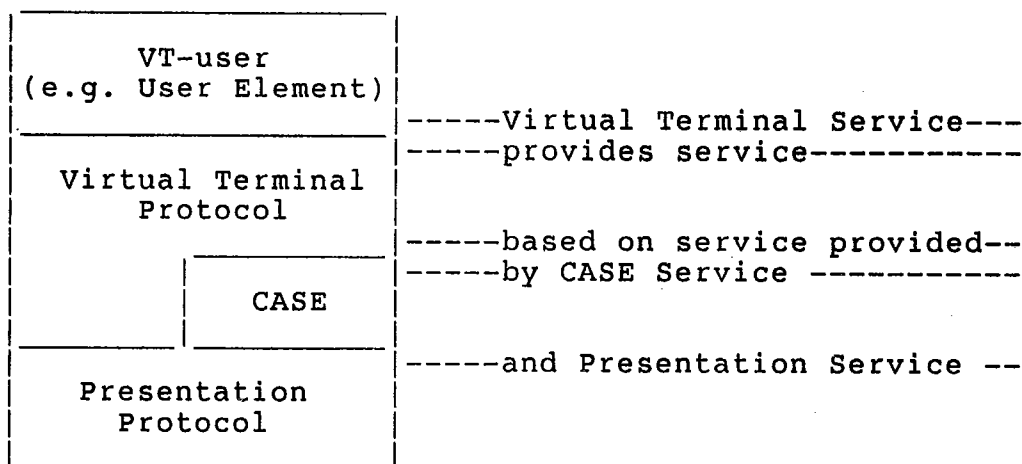


Figure 1 - Relationship of this Standard to other OSI Application Layer Standards

1 SCOPE AND FIELD OF APPLICATION

1.1 Scope

This International Standard defines, in an abstract way, the externally visible Basic Class Virtual Terminal Service within the OSI Application Layer in terms of:

- a) a model defining the interaction between users of the service
- b) the primitive actions and events of the service
- c) the parameter data associated with each primitive action and event
- d) the relationship between, and the valid sequences of, these actions and events.

The service defined in this International Standard is that which is provided by the OSI Basic Class Virtual Terminal Protocol (in conjunction with the Common Application Service Element and the Presentation Service) and which may be used by any User Element or other Application Service Element.

This International Standard also defines two default profiles in Annex A and a number of draft profiles in Annex B. These profiles define sets of virtual terminal environment parameters for use in the establishment of virtual terminal associations and subsequent negotiation.

This International Standard does not specify individual implementations or products, nor does it constrain the implementation of entities and interfaces within a computer system. There is, therefore, no requirement for conformance to this International Standard.

1.2 Field of Application

This International Standard applies to interactive applications requiring terminal oriented communication expressed in terms of the transmission and manipulation of graphical images having the following characteristics:

- a) the images are composed of character-box graphic elements organised into a one, two or three dimensional structure;
- b) attributes may be associated with any graphic element to qualify its mode of display.

2 REFERENCES

ISO 646 Information Processing - ISO 7-bit Coded Character Set for Information Exchange.

ISO 2022 Information Processing - ISO 7-bit and 8-bit Coded Character Sets - Code Extension Techniques.

ISO 2375 Data Processing - Procedure for Registration of Escape Sequences.

ISO 6429 Information Processing - ISO 7-bit and 8-bit Coded Character Sets - Additional Control Functions for Character Imaging Devices.

ISO 7498 Information Processing Systems - Open Systems Interconnection - Basic Reference Model.

ISO 8649/2 Information Processing Systems - Open Systems Interconnection - Common Application Service Elements - Part 2. Association Control. [1]

ISO 8824 Information Processing Systems - Open Systems Interconnection - Specification for Abstract Syntax Notation One (ASN.1).

TR 8509 Information Processing Systems - Open Systems Interconnection - Service Conventions. [1]

The International Register of Coded Character Sets to be used with Escape Sequences. [2]

[1] At present at the stage of draft.

[2] Available from the European Computer Manufacturers Association (ECMA), 114 rue du Rhone, CH-1204 Geneva, Switzerland.

[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 International Standard 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) service data unit;
- d) user element.

It also makes use of the following terms defined in TR 8509:

- e) service
- f) service primitive;
- g) service provider;
- h) primitive;
- i) request (primitive);
- j) indication (primitive);
- k) response (primitive);
- l) confirm (primitive);
- m) confirmed service;
- n) non-confirmed service;
- o) provider-initiated service;
- p) service access point.

3.2 Common Application Service Element Definitions

This International Standard makes use of the following terms defined in ISO 8649/2:

[ISO page 4]

- a) application association;
- b) application entity title.

3.3 Virtual Terminal Service Definitions

3.3.1 VT-user: A user of the Virtual Terminal Service.

3.3.2 Repertoire: A set of up to four Coded Character Sets, as defined in ISO 2022, and designatable by the Escape Sequences of ISO 2022 in conjunction with ISO 2375 and the International Register of Coded Character Sets to be used with Escape Sequences. The elements of the sets forming a Repertoire constitute the set of objects which may individually occupy, as primary attribute values, the array elements of a Display Object when that Repertoire is in use.

3.3.3 Character-box Graphic Element: An atomic element of a repertoire where use of the repertoire has been agreed through negotiation by the VT-users.

3.3.4 Display Object: An abstract object, defined in this standard, for the exchange of graphic information. It consists of a one, two or three dimensional array of character-box graphic elements.

3.3.5 Control Object: An abstract object, of a type defined in generic terms in this Standard, for the exchange of unstructured information of a single type.

NOTE -- The primary application of a Control Object is for the exchange of information of a control nature, as understood by the VT-users, but the VT Service applies no constraint on the nature of the information.

3.3.6 Object Updating Device: A real device capable of generating values which (possibly after undergoing a transformation) are used by one of the peer VT-users to update either a display object or control object (or possibly both).

[ISO page 5]

3.3.7 Termination-event: A locally defined atomic event related to input from an object updating device (e.g. entering a character, pressing a function key, setting a flag) which a VT-user chooses to designate as causing input data to be delivered to the peer VT-user.

3.3.8 VT-association: An application association between two peer VT-users.

3.3.9 VT-environment (VTE): A set of parameters that together define the data structuring and operational characteristics for a particular VT-association. The VTE exists only during the lifetime of that VT-association. The parameters of the set are mutually related by a directed graph structure. The VTE may be modified during the existence of the VT-association by negotiation.

3.3.10 Current-VTE: The single VTE that exists for a VT-association when negotiation is not in progress.

3.3.11 Draft-VTE: The VTE, if any, under negotiation. It may be incomplete during the negotiation.

3.3.12 VTE-parameter: An individual parameter of a VTE.

3.3.13 Full-VTE: A VTE that is a complete directed graph of VTE-parameters in which all node parameters and terminal leaf parameters implied by all existing nodes from the root of the tree have values.

3.3.14 VT-context-value: A collective term for the set of object instances, their assigned values and the current-VTE for a particular VT-association. A VT-context-value exists only during the life time of the VT-association and is normally changing continuously during this time interval.

3.3.15 Access-right: An abstract object that confers on the owner the right to perform any of a particular set of operations that are defined as being controlled by the access-right. An access-right may be reassignable, (i.e. ownership of it may be transferred from one owning entity to another, previously non-owning, entity), or non-reassignable, (i.e. one entity owns it for the whole duration of its existence that is generally coterminous with the VT-association).

3.3.16 Net-effecting: The conversion of a sequence of operations and/or items of information which update a set of objects into a different, usually shorter sequence which results in the same final states of the objects.

3.3.17 A-mode (Asynchronous mode): A mode of operation utilising two monologues (in opposing directions) to provide update access to two display objects, one monologue for each display object; update access to each display object is controlled by a different non-reassignable access-right.

3.3.18 S-mode (Synchronous mode): A mode of operation utilising one two-way alternate dialogue supporting one display object; update access to the display object is controlled by a single reassignable access-right.

3.3.19 Service: A distinct part of the total VT Service that is composed of a sequence of primitives taken from the set {request primitive, indication primitive, response primitive, confirm primitive}.

3.3.20 Sequenced Service: A Service for which an indication (or confirm) primitive resulting from a corresponding request (or response) primitive is initiated in sequence with all previously initiated indications (or confirms) and their corresponding requests (or responses).

3.3.21 Non-sequenced Service: A Service for which an indication (or confirm) primitive resulting from a corresponding request (or response) primitive is not necessarily initiated in sequence with all previously initiated indications (or confirmations) and their corresponding requests (or responses).

3.3.22 Conditionally Sequenced Service: A Service for which

- a) certain values for parameters of the primitives result in sequenced operation, and
- b) other values for parameters of the primitives result in non-sequenced operation.

3.3.23 Destructive Service: A service that may cause the loss of information conveyed in previously initiated services without notification of this loss to either VT-user. Only non-sequenced services may be destructive, but not all non-sequenced services are destructive.

3.3.24 Non-destructive Service: A service that does not cause the loss of information conveyed in previously initiated services without notification to the VT-users.

3.3.25 Service Parameter: A parameter defined as part of a primitive within a VT service.

NOTE -- The service parameters are defined for each VT service in Section Four.

3.3.26 Update-window: An optional addressing mechanism associated with display object addressing which may be used to define a range of coordinate values for each dimension below which an update operation may not be performed.

3.3.27 Update-window-size: A positive, non-zero integer that defines the number of contiguous array elements within a update-window.

3.3.28 Trigger: A mechanism associated with a control object that automatically causes delivery of updates when the contents of the control object are updated. The trigger mechanism also causes reassignment of the WAVAR access-right in S-mode.

3.3.29 Profile: A pre-defined set of VTE-parameter values making up a full-VTE.

NOTE -- Some profiles are parameterised in that values for profile parameters must be supplied by the VT-users.

3.3.30 Default Profile: A profile, defined in Annex A, that is used to establish a full-VTE in the absence of a VT-user specified profile at VT-association establishment.

3.3.31 Registered Profile: A profile registered in the VT Profile Register of the ISO Registration Authority.

NOTE -- At the time of the DIS, this Register and Registration Authority do not exist. It is expected that, at some time in the future, the Registration Authority will be established under the authority of the International Organization for Standardization (ISO). When such an authority is established, it is anticipated that a finalised version of the draft profiles in Annex B will be transferred to the profile register maintained by that Registration Authority.

3.3.32 Privately Defined Profile: A profile that is neither a default profile nor a registered profile whose use has been agreed privately outside of the scope of this standard.

3.3.33 Profile Parameter: A parameter of a VT service profile. It may correspond exactly to a VTE-parameter or may be a special profile parameter. Each registered profile definition defines the applicable profile semantics.

NOTE -- Agreement by the VT-users of a value for all profile parameters is required before a full-VTE may be constructed for use.

3.3.34 Special Profile Parameter: A profile parameter which is not a VTE-parameter (i.e. is not from the directed graph) but has a valid use as defined in the profile definition. Values for special-profile-parameters are always one of the ASN.1 primitive types BOOLEAN, INTEGER or PrintableString.

3.3.35 MIN Initiator: The VT-user that initiated the VT-START-NEG service which was successful in causing the transition to Negotiation Active phase.

3.3.36 MIN Acceptor: The VT-user that accepted the VT-START-NEG service which was successful in causing the transition to Negotiation Active phase.

4 ABBREVIATIONS

4.1 General

ISO	International Organization for Standardisation
OSI	Open Systems Interconnection
CASE	Common Application Service Elements
VT	Virtual Terminal
VTE	Virtual Terminal Environment
MIN	Multiple Interaction Negotiation
IRV	International Reference Version (of ISO 646)

4.2 Modes of operation

A-Mode	Asynchronous Mode
S-Mode	Synchronous Mode

4.3 VTE model components

ACS	Access Control Store
CCA	Conceptual Communication Area
CDS	Conceptual Data Store
CO	Control Object
CSS	Control, Signal and Status store
DSD	Data Structure Definition
DO	Display Object
TCO	Termination Control Object

4.4 Access-rights

WACA	Write Access Connection (VT-association) Acceptor
WACI	Write Access Connection (VT-association) Initiator
WAVAR	Write Access VARIABLE

5 CONVENTIONS

This International Standard uses the descriptive conventions contained in the ISO Service Conventions (TR 8509).

Characters inclosed in angle brackets (i.e., "<", ">") denote the equivalent ISO 646 mnemonic.

SECTION ONE -- OVERVIEW6 GENERAL FEATURES6.1 Introduction

The Basic Class Virtual Terminal Service supports the interactive transfer and manipulation of graphic data by VT-users. This graphic data is structured in a manner which models the class of character-box oriented terminals. The structuring of graphic elements is limited to images consisting of character-box graphic elements arranged in a one, two or three dimensional array.

Two modes of operation are defined for Basic Class, S-mode and A-mode.

6.2 Features of the Service

The Basic Class Virtual Terminal Service offers the following services to the VT-user:

- a) the means to establish a VT-association between two peer VT-users for the purpose of enabling virtual terminal information exchange;
- b) the means to negotiate a service subset and a consistent set of VTE-parameters;
- c) the means to transfer and manipulate structured data in a way that is independent of the local representation of information used by each VT-user and that is independent of the way in which supporting communications media are used;
- d) the means to control the integrity of the communication;
- e) the means to terminate the VT-association either unilaterally or by mutual agreement;
- f) the means to support both synchronous and asynchronous modes of operation between the VT-users.

6.3 VTE Parameters

The transfer and manipulation of graphic data takes place within a VT-environment (VTE) defined by a logically consistent set of VTE-parameter values. The logical relationships between VTE-parameters determine the constraints on the values which may be assigned to related VTE-parameters.

One and only one full-VTE exists during data transfer. A full-VTE may be modified or replaced if negotiation facilities are available.

6.4 VT Profiles

The Basic Class Virtual Terminal Service provides for the use of VT profiles which are sets of predefined VTE-parameters for use in the negotiation of the VTE (see clause 11 and annexes A and B).

6.5 Dialogue Control

The integrity of VT communication is maintained by dialogue control service facilities (involving access-right mechanisms, see clause 9) and service primitive sequencing rules which provide integrity by detecting and resolving collisions (see clause 24).

7 COMMUNICATION FACILITIES

7.1 Establishment Facility

The establishment facility provides a service that establishes a VT-association and an initial VTE for that VT-association when a VT-user first invokes the Basic Class Virtual Terminal Service. It either initialises an existing application association (received from the application entity) or, if an application association does not already exist, it establishes and initialises a new application association for the purpose of VT communication.

7.2 Termination Facility

The termination facility provides services which enable a VT-user to terminate a VT-association either in an orderly and non-destructive manner or in an immediate and potentially destructive manner. It also provides a service which enables the Virtual Terminal Service to notify the VT-users when a VT-association is terminated in a potentially destructive way by the VT service provider.

7.3 Negotiation Facility

The negotiation facility provides services which enable peer VT-users to select, modify and replace the VTE.

An initial VTE is established during VT-association establishment based on the VT profile specified. If the VT-user does not specify a profile during VT-association establishment, a default profile is used to establish the initial VTE. This VTE may subsequently be modified or replaced depending on the type of negotiation facilities available.

The type of negotiation facility available to the VT-users is determined by the service subset selected during VT-association establishment. In addition to the profile selection facility available during VT-association establishment, there are two other types of negotiation, i.e. switch profile negotiation and multiple interaction negotiation.

7.4 Data Transfer Facility

The data transfer facility provides a service which enables a VT-user to transfer updates to a display object or a control

object in a manner that preserves the updates' semantic content. The VT-user's ability to transfer updates is dependent on the ownership of the appropriate access-right.

7.5 Delivery Control Facility

The delivery control facility provides services which enable a VT-user to control, synchronise and optionally request acknowledgement of, the release of updates previously transferred to the peer VT-user.

7.6 Access-right Management Facility

The access-right management facility provides services which enable the VT-users to control the ownership of any reassignable access-rights which mediate the use of other facilities. These services are applicable only when the "WAVAR" access-right is defined, i.e. in S-mode.

SECTION TWO -- GENERAL8 MODES OF OPERATION

The Basic Class Virtual Terminal Service supports two modes of operation. Each mode is characterised by its form of dialogue control. The mode of operation is fixed for a VT-association and is not explicitly declared but is determined by the access-rights negotiated for the VT-association.

8.1 S-mode

S-mode (synchronous mode) has the following characteristics:

- a) S-mode operates when the WAVAR access-right has been selected;
- b) S-mode is limited to a single two-way dialogue;
- c) S-mode supports only a single display object (see clause 9); update access to this display object is controlled by the WAVAR access-right;
- d) ownership of the WAVAR access-right may be passed between the two VT-users;
- e) the VT-users may define and make use of termination condition parameters in device objects;
- f) control objects may be defined to have update access controlled by the WAVAR access-right

8.2 A-mode

A-mode (asynchronous mode) has the following characteristics:

- a) A-mode operates when the WACI & WACA access-rights have been selected;
- b) A-mode is limited to two monologues in opposing directions;
- c) these monologues support update access to two display objects (see clause 9), one for each direction;

[ISO page 16]

Documents

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

- d) update access to each of these display objects is controlled by a non-reassignable access-right for each display object which is established during VTE establishment;
- e) the VT-users may define and make use of echo control objects and termination condition parameters in device objects;
- f) control objects may be defined to have update access controlled by WACA or WACI access-rights (see clause 9).
- g) access-right management service primitives do not apply when operating in A-mode.

9 ACCESS-RIGHTS

An access-right is an abstract object representing permission to carry out some set of operations. Ownership of an access-right confers permission to perform the defined operations.

The selection of which access-rights are available during a particular instance of communication is made during VT-association establishment. This selection determines the mode of operation and is either

- a) a single reassignable access-right (S-mode)

WAVAR (Write Access VARIABLE), whose ownership can be passed between the two VT-users and whose initial ownership is agreed between the VT-users when the VT-association is initialised; or

- b) a pair of non-reassignable access-rights (A-mode)

1) WACI (Write Access Connection Initiator), owned at all times by the VT-user that initiated the VT-association;

2) WACA (Write Access Connection Acceptor), owned at all times by the VT-user that accepted the VT-association.

The access-rights control update access to

- c) display objects (see clause 13 and 17.1); and

- d) control objects with the "controlled" attribute (see clause 14 and 19.1).

If the WAVAR access-right is selected (S-mode), the single display object always has the WAVAR access-right and "controlled" control objects may only be assigned the WAVAR access-right.

If the WACI and WACA access-rights are selected (A-mode), one display object has the WACI access-right and the other display object has the WACA access-right. "Controlled" control objects may be assigned either the WACI or WACA access-rights.

The access-rights for any control objects created via negotiation subsequent to VT-association establishment are constrained to follow these rules.

In S-mode, communication facilities allow a VT-user to request or cede ownership of the WAVAR access-right.

Table 1 shows the valid combinations of VT access-rights for each facility (see clause 7) and mode of operation.

Table 1 -- VT Access-right and Mode/Phase Correspondence

FACILITIES \ MODE	MODE	
	S-MODE	A-MODE
NEGOTIATION	WAVAR	(NONE)
DATA HANDLING	WAVAR	WACI & WACA
TERMINATION	WAVAR	(NONE)

10 SUBSETS AND NEGOTIATION

10.1 Service Subsets

Three service subsets, VT-A, VT-B and VT-C, are defined for the Basic Class Virtual Terminal Service. They differ only in the level of negotiation available in each subset. Each successive subset completely contains all the capabilities of the previous subset. Each subset applies equally in each mode of operation. Table 2 specifies the service facilities present in each subset.

Table 2 -- Service Subsets

Subsets	VT-A	VT-B	VT-C
Facilities	Kernel Subset	Switch Profile Negotiation Subset	Full Negotiation Subset
Establishment	present	present	present
Termination	present	present	present
Switch Profile Negotiation		present	present
Multiple Interaction Negotiation			present
Data Transfer	present	present	present
Delivery Control	present	present	present
Access-right Manage- ment (S-mode only)	present	present	present

10.2 Principles of Negotiation

The Basic Class Virtual Terminal Service provides negotiation services to enable the creation and modification of the single VTE in a manner acceptable to the service provider and both VT-users. Negotiation is available as part of the establishment facility. Additional negotiation services are optionally available (dependent on the subset selected during establishment) which provide two forms of negotiation, a single

interaction switch profile negotiation and a multiple interaction negotiation.

At VT-association establishment, a current-VTE is established using the profile based negotiation function embedded in the VT-ASSOCIATE service. This current-VTE may or may not be a full-VTE. If the initiating VT-user does not specify a profile with the VT-ASSOCIATE (to supply the basis for negotiating the current-VTE), the service selects a default profile appropriate to the mode of operation, see Annex A. Depending on the service subset, this current-VTE may or may not be subsequently modified using negotiation services. If the initial current-VTE is not a full-VTE, it is necessary to use the negotiation facility to establish a full-VTE before the Data Handling Phase may be entered.

The negotiation services do not modify the current-VTE unless the negotiation reaches a satisfactory conclusion. During an instance of negotiation, agreements on VTE-parameter values are recorded in a temporary VTE definition known as the draft-VTE. In the case of single interaction profile switch negotiation, a draft-VTE only has a transitory existence. However, for multiple interaction negotiation, the concept is more significant, i.e. the draft-VTE contains the new VTE as it is built up over an extended time interval.

10.2.1 Switch Profile Negotiation

The single interaction switch profile negotiation is provided by a single confirmed service, VT-SWITCH-PROFILE. Negotiation takes the form of a proposal made by one VT-user using a VT profile and, where required, offered profile parameter values. The profile chosen is not negotiable but offered profile parameter values may be adjusted by both the service provider and the accepting VT-user (in that order) to obtain an agreed set of VTE-parameter values, but only within the range of freedom offered by the service initiator. Adjustment of VTE-parameter values by this service is constrained to those parameters defined as profile parameters in the profile specification.

10.2.2 Multiple Interaction Negotiation

Multiple interaction negotiation (MIN) may only be initiated and terminated by confirmed services, ensuring the agreement of both the VT-users and the service provider to the proposed action.

MIN is provided to enable the negotiation of a set of VTE-parameter values comprising a full-VTE in a series of steps (the intermediate steps do not necessarily form a complete or consistent full-VTE).

MIN negotiation is supported by four services:

- a) INVITE: a VT-user invites the peer VT-user to propose values for VTE-parameters;
- b) OFFER: a VT-user proposes values or value ranges to the peer VT-user; this may be in response to an INVITE or as a counter-offer to a previous OFFER;
- c) ACCEPT: a VT-user accepts VTE-parameter values proposed by the peer VT-user in an OFFER;
- d) REJECT: a VT-user rejects VTE-parameter values proposed by the peer VT-user in an OFFER.

Using these services, independent MIN sequences may be performed on different VTE-parameters. Valid MIN sequences and the rules for use of the MIN services are defined in 27.3.

NOTE -- The stages of a MIN negotiation may be independent or may depend on the results of earlier stages.

11 SERVICE PROFILES

A service profile is a logically consistent set of VTE-parameter values defining a VTE. In some profile definitions, leaf VTE-parameters may be left without explicit values and other options may be provided using the technique of profile parameters. In order to construct a full-VTE from such a profile, explicit values must be negotiated for these profile parameters by the VT-users and the service provider either during VT-association establishment or subsequent negotiation.

VT profiles also define control object and device object semantics.

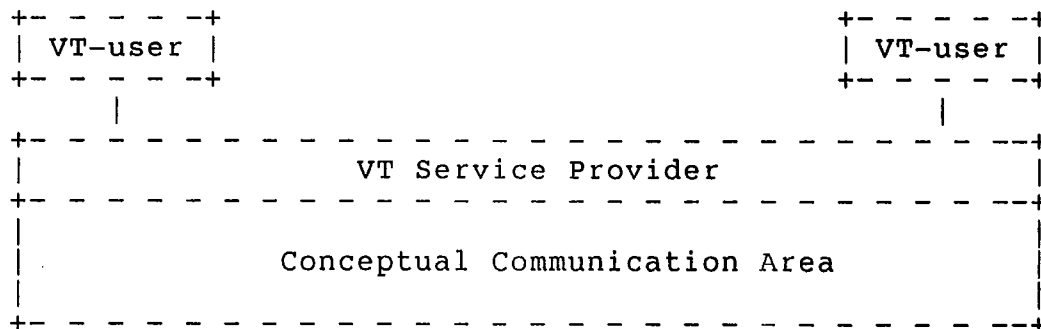
Two default profiles and a number of other profiles are defined for this service. The default profiles, one for S-mode and the other for A-mode, are defined in Annex A and the other profiles are contained in Annex B. Private profiles may be defined outside the scope of this International Standard.

NOTES

1. Profiles provide an efficient method for defining the values of a number of VTE-parameters simultaneously. They also provide a means of promoting interoperability by registering well known profiles which have been designed for a particular purpose. See Annex B for a full description of the conventions for defining VT profiles and profile parameters.
2. It is expected that the profile definitions in Annex B will be removed from this Standard when a Register of VT Profiles has been established.

SECTION THREE -- VT ENVIRONMENT12 THE VTE MODEL12.1 Structure

VT-users communicate by means of a shared conceptual communication area (CCA) containing a number of abstract objects and object type definitions. Information exchange is modelled by one VT-user updating the contents of the CCA, the changed state of the CCA then being made accessible to the peer VT-user. The VT-users update the CCA via the services provided by the service provider (see figure 2).

Figure 2 -- Service Model

The CCA containing the abstract objects used for describing the VT service has no physical existence.

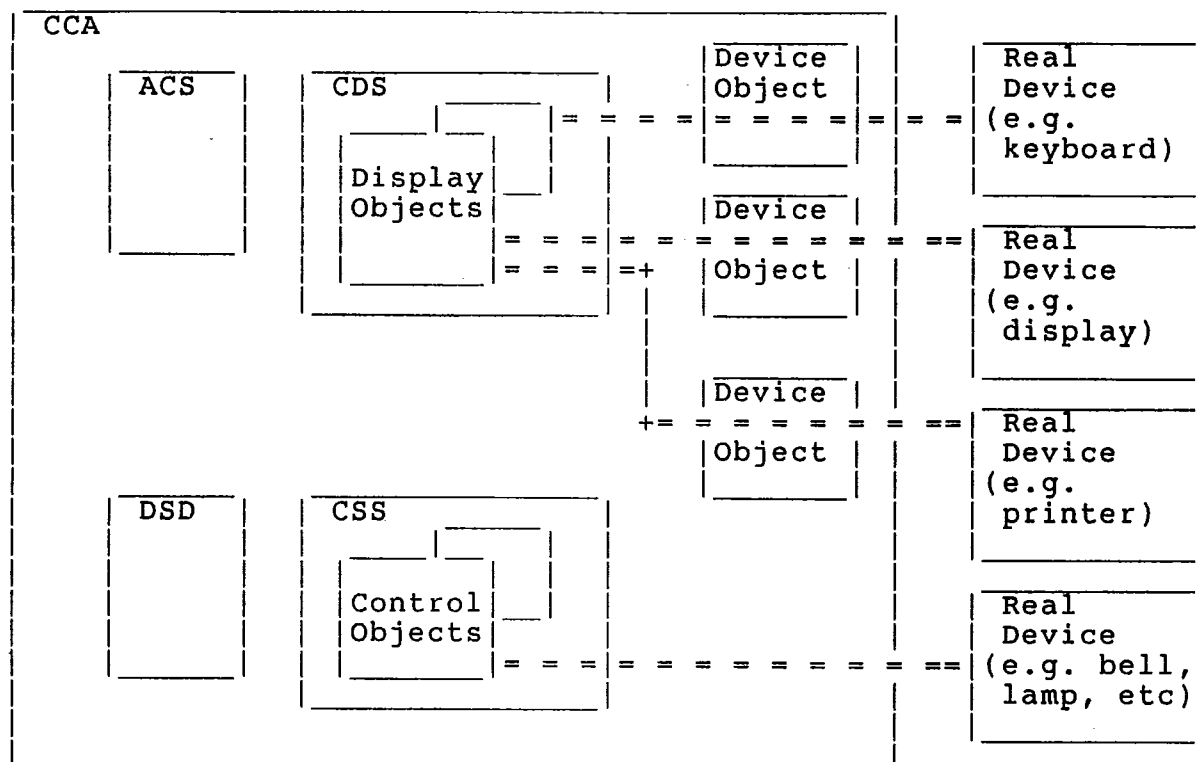
The CCA is partitioned into four sub-areas:

- a) a conceptual data store (CDS) containing one or two display objects;
- b) a control, signalling and status store (CSS) containing zero or more control objects which may be used not only for device control but also for signalling and status information and for any other purposes agreed outside the scope of the standard;
- c) an access control store (ACS) which records the current assignment of all applicable access-rights;
- d) a data structure definition (DSD) that contains the object type definitions for the display objects, device objects, control objects (see clauses 13, 14 and 15) and other negotiated VTE-parameters which form the principal part of the current-VTE definition.

[ISO page 23]

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

Figure 3 illustrates the partitioning of the CCA and indicates how display objects and control objects are mapped onto real devices via the related device objects.



NOTE -- This is an example of an A-mode mapping. The mapping of Objects to Real Devices is represented by = = = = =

Figure 3 -- CCA Partitioning

The contents of the DSD are initialised to correspond to the current-VTE at VT-association establishment time and may only be changed subsequently by negotiation. The contents of the remaining sub-areas are open to change when the VT-association is in existence and negotiation is not in progress.

A VT-user requires ownership of the appropriate access-rights in order to modify those objects in the CDS and CSS which are subject to access control (see clause 9). The VT-user may synchronise and control updates to a set of CCA objects which have the same access-right using the delivery control facility.

NOTE -- Section Four defines the communication facilities and the primitives which VT-users use to modify the CCA. This standard does not preclude local mechanisms or "macro" operations which combine primitives at a local interface.

13 DISPLAY OBJECTS

This standard does not define the semantics associated with the structure of the display object(s), nor the relationship between this structure and the real device. Such semantics, i.e. the meaning of the update operations which are performed on the display object structure, depends on the profile used (see C.11).

13.1 Structure

A display object is a one, two or three dimensional array of array elements each capable of holding one character-box graphic element from a repertoire of such elements as defined in the VTE (see 17.2.2).

The dimensions are named X, Y, and Z. X is the lowest order dimension. A set of array elements identified by a contiguous set of x-coordinate values is known as an X-array. Y is the next higher order dimension. A set of contiguous y-coordinate values identifies a set of X-arrays known as a Y-array. Z is the highest order dimension. A set of contiguous z-coordinate values identifies a set of Y-arrays known as a Z-array (see also C.11).

Each array dimension has a lower bound of one. An upper bound may be defined or the array dimension may be declared unbounded. There are no restrictions on combinations of bounded and unbounded dimensions for multi-dimensional arrays.

This structure is addressed by means of a display pointer that uniquely identifies a single array element; it consists of a set of array coordinate values with one value for each defined array dimension. See also C.3.

13.2 Attributes

At any given time, an array element is either empty (i.e. it has no primary attribute value currently assigned to it) or it contains a primary attribute value selecting a character-box graphic element from a repertoire of such elements.

In addition, each array element contains secondary attribute values specifying the repertoire in use and display rendition attributes for that array element. The rendition attributes that may be specified (once their use has been negotiated) are emphasis, foreground colour, background colour and font. Each array element, with its content of primary and secondary

[ISO page 25]

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

attributes, is self-contained and completely independent of all other array elements. It is not necessary that an array element has a primary attribute value assigned in order to have secondary attribute values assigned.

NOTES

1. The interpretation of any empty array element, and in particular its physical representation on displays or printers, is not defined by this standard.
2. The effect of display rendition attributes on an empty array element is not defined in this Standard.

14 CONTROL OBJECTS

Control objects enable VT-users to handle control information related to virtual terminal functions and to real devices. The content of the information field is limited (see clause 19) and the semantics of this information is defined in a profile, defined as part of a registered control object, or made known to both VT-users by means outside the scope of this International Standard.

Control objects are selected in the initial VTE when establishing a VT-association if they exist in the selected profile.

Subsequent negotiation may cause a different set of control objects to be selected if a new profile is used. Negotiation may also define new control objects using VTE-parameters (see clause 19) or alter the characteristics of existing ones.

A control object may be linked to a device object (see clauses 15 and 20).

A control object has a single unique name. This name is used to identify the control object when its information field is to be updated. It is also used to relate a registered control object to the register entries.

A control object may be given an access-right from the access-rights available for the VT-association. Those control objects subject to access control may also be defined as having a trigger mechanism assigned to them. If the access-right is reassignable (i.e. WAVAR), then the trigger mechanism ensures that the WAVAR access-right is automatically reassigned to the peer VT-user when the control object is updated (see also clause 21 on delivery control). If the access-right is not reassignable, the trigger mechanism is confined to delivery of updates to all objects controlled by the particular access-right.

A default control object is implicitly defined when each device object is defined. This default control object contains eight booleans and may be defined as having a trigger mechanism, see C.4.

NOTES

1. Control objects may be associated with device objects or may have an independent existence in the CCA. The handling of information of a control nature associated with a device which requires more than the eight booleans defined for the default control object associated with the device object is an example of where

[ISO page 27]

an additional control object may be specifically designed to handle the types of status information to be transferred.

2. Some control objects are defined in detail in this Standard and other control objects may be the subject of registration, either individually or in VT profiles.

15 DEVICE OBJECTS

Device objects provide VT-users with a mechanism for specifying certain real device related characteristics to aid in mapping information contained in display objects to/from real devices. Device objects also serve to relate devices to display objects and any control objects which may have been selected as part of a VTE (see clause 14) to assist in the operation of the device.

Device objects are selected in the initial VTE when establishing a VT-association when they exist in the selected profile.

Subsequent negotiation may cause a different set of device objects to be selected if a new profile is used. Negotiation may also define new device objects using VTE-parameters (see clause 20) or alter the characteristics of existing ones.

A device object has a single unique name.

Device objects are not updated directly using the data transfer facility. The VTE-parameters of a device object are intended to influence the mapping between a display object and a real device.

A default control object is associated with each device object. This control object does not use the normal control object VTE-parameters (clause 19) but has pre-defined characteristics and certain specific VTE-parameters in the device object (see clause 20). For update purposes, this control object is referenced by the name of the device object (no other use is made of the device object name in the VTS).

[ISO page 29]

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

16 DIRECTED GRAPH OF VTE PARAMETERS

The VTE-parameters are structured as a directed graph as specified in figure 4. The figure also indicates those VTE-parameters which may have multiple occurrences.

A full-VTE is required to have values for all VTE-parameters of the directed graph,

- a) including values for multiple occurrences that are defined by the values given to higher order (parent) VTE-parameters in the directed graph, but
- b) excluding values for VTE-parameters whose non-existence is implied by the values given to higher order (parent) VTE-parameters in the directed graph.

The individual parameters of the directed graph are defined in clauses 17, 19, 20 and 21 except for VT-access-rights and VT-class-subset which are not VTE-parameters but are service parameters of the VT-ASSOCIATE service, (see 25.1.4).

[illegible]

Note: The notation below indicates this parameter may have multiple occurrences:

```

- - - - - vte-parameter-name

```

17 DISPLAY OBJECT VTE-PARAMETERS

This clause defines the VTE-parameters for the display objects. These VTE-parameters have their values assigned through negotiation either at VT-association establishment time or by explicit negotiation subsequent to VT-association establishment. Their values determine the display characteristics of the VT service.

17.1 Primary VTE-Parameters

The primary characteristics of a display object are defined by the VTE-parameters in table 3. These VTE-parameters establish the structure of the display object and the set of attributes that may be associated with each array element.

Table 3 -- Primary VTE-Parameters

Parameter	Value
display-object-name	character string of type ASN.1 Printable String
object-access-right	"WACI", "WACA", "WAVAR" (see clause 9)
dimensions	"one", "two", "three" (default="two")
erasure-capability	"yes", "no" (default="no")
repertoire-capability	1..N (default=1)
emphasis-capability	1..N (default=1)
foreground-colour- capability	1..N (default=1)
background-colour- capability	1..N (default=1)

The erasure-capability VTE-parameter controls the availability of the erase-X-array (18.2.3), erase-Y-array (18.2.4) and erase-Z-array (18.2.5) update operations for the display object. The value "yes" indicates all of them are available (subject to the display object structure and addressing rules); "no" indicates none of them are available.

The VTE-parameters repertoire-capability, emphasis-capability, foreground-colour-capability and background-colour-capability specify the number of attributes or attribute sets that may be defined by the corresponding assignment VTE-parameters defined in 17.2.2 and 17.2.3.

NOTE -- These VTE-parameters do not themselves define the individual attribute values. For example, the emphasis-capability value represents the number of different emphasis-assignment values which may be used.

17.2 Secondary VTE-Parameters

Certain of the VTE-parameters defined in this subclause and in 17.3 may have multiple occurrences in ordered lists. During update operations, a value is selected by its position in the ordered list. For each ordered list, there is a higher order VTE-parameter of the "capability" type which determines the maximum number of such occurrences, see 17.1.

In the case of the multiple occurrence VTE-parameters for repertoire, emphasis and colour assignment in 17.2.2 and 17.2.3, the first entry in the ordered list is the explicit modal default value. In the case of the font-assignment VTE-parameter in 17.3, the first entry in the list corresponding to the current modal value for repertoire assignment is the explicit modal default. If the ordered list for one of these VTE-parameters is absent, the implicit default (defined in table 6 or 7) applies.

17.2.1 Addressing VTE-Parameters

The VTE-parameters defining the display object addressing constraints are specified in table 4. They are always present if the corresponding dimension has been defined.

Table 4 -- Addressing VTE-parameters

Parameter	Value
x-bound	"unbounded", 1..N (default="unbounded")
x-forward	"no constraint", maximum increment 0..N; value 0 implies no forward relative move is permitted; (default="no constraint")
x-backward	"no constraint", maximum decrement 0..N; value 0 implies no backward relative move is permitted; (default=0)
x-absolute	"no constraint", "higher only", "not permitted" (default="not permitted")
x-window	"unbounded", 1..N (default=1 for the case where x-bound = "unbounded" [i.e. no backward motion is permitted]; otherwise "unbounded")
y-bound	Values as above, but with substitution of "y" for the "x" dimension references
y-forward	
y-backward	
y-absolute	
y-window	
z-bound	Values as above, but with substitution of "z" for the "x" dimension references
z-forward	
z-backward	
z-absolute	
z-window	

The x-bound VTE-parameter defines the upper bound of the x dimension.

The x-forward and x-backward VTE-parameters define relative addressing constraints; maximum-increment and maximum-decrement values give the maximum change which may be made to the coordinate value of the dimension by a single relative addressing operation in, respectively, the forward (increase in value) and backward (decrease in value) directions.

The x-absolute VTE-parameter defines absolute addressing constraints; the value "higher only" constrains an absolute addressing operation to be such that the coordinate value for the dimension may only be increased (or remain unchanged).

The x-window VTE-parameter defines the update-window size for the x dimension.

The values of these secondary VTE-parameters are not interdependent.

17.2.2 Repertoire-assignment VTE-Parameters

These VTE-parameters enable a repertoire or repertoires to be assigned and a set of font styles to be associated with each such repertoire.

A repertoire (see C.12) is composed of the following components as defined in ISO 2022:

- a) a C0 set, optional;
- b) a C1 set (see C.13), optional;
- c) a G set usable in the GL position of the code table, mandatory;
- d) a G set usable in the GR position of the code table, optional.

Alternatively, a repertoire may consist only of the Transparent Coded Character Set. This is a 256 element set specially registered for this Standard. It consists of 256 values for 0..255 with no assigned meanings in the VT Service. It has the ISO Registration Number of 125. [1]

Inclusion in a repertoire of a G-set which includes non-spacing diacritical marks implies that the valid combinations of these with other graphic characters, as defined in the registered code set, are available in the repertoire as storable

[1] At the time of the DIS, application for the registration of the Transparent Coded Character Set had been made but had not been confirmed.

characters occupying single array elements. Support of such storage is implied by acceptance of such a repertoire. No other combinations using the non-spacing marks are available, nor can they be used alone -- use with space may be available in the character set definition.

If a repertoire includes a G-set which is defined in the register as a multi-byte set, this implies that the repertoire includes all the valid combinations as defined in the register, as separate storable characters occupying single array elements. Support of such storage is implied by acceptance of such a repertoire. No other combinations are permitted.

The VT Service does not support, as part of a repertoire, compound characters formed using the <BS> (i.e. backspace) control of ISO 646 to achieve super-imposition of two graphic characters, see C.12 and C.13.

If the mandatory G set is a 94 character set, it is implicit that the SPACE character is available in position 2/0 of the GL code table.

The two VTE-parameters specified in table 5 may be defined once for each repertoire indicated as available by the integer value of the VTE-parameter repertoire-capability. Multiple occurrences are treated as an ordered list where individual occurrences (i.e. repertoires) are referenced by their positions in the ordered list.

Table 5 -- Repertoire VTE-parameters

Parameter	Value
repertoire-assignment	Each instance of this VTE-parameter is itself a list; this list is a list of ISO 2022 defined character set designating escape sequences or the symbolic value "void" and defines a repertoire. The list (repertoire) may contain one to four escape sequences which define, in order, the GL set, the GR set (or "void"), the C0 set (or "void"), and the optional C1 set (default = <ESC> 2/8 4/0 [the set of graphic characters of the IRV of ISO 646]; this is a repertoire consisting of a single graphic character set located in the GL set of the code table). Alternatively, the list may consist of the single escape sequence <ESC> 2/5 2/15 4/2 for the transparent repertoire.
font-capability	Each instance of this VTE-parameter is an integer 1..N specifying the number of fonts available for the repertoire with which this VTE-parameter is associated (default=1 for each implied occurrence).

NOTES

1. For the meanings of these terms, see ISO 2022, which also gives the general rules for use of registered code sets. For the definition of the character sets designated by particular escape sequences, refer to the International Register of Coded Character Sets to be Used with Escape Sequences, which also gives any particular rules for the use of a set.
2. Registration Number refers to the numbers assigned by the registration authority in the International Register of Coded Character Sets to be Used with Escape Sequences.
3. The list of fonts available for each repertoire is specified by the VTE-parameter font-assignment, see 17.3.

17.2.3 Emphasis and Colour "assignment" VTE-Parameters

Multiple instances of the following VTE-parameters may be defined,

- a) emphasiss-assignment VTE-parameter;
- b) foreground-colour-assignment VTE-parameter;
- c) background-colour-assignment VTE-parameter.

The values for these VTE-parameters are specified in table 6.

The maximum number of instances for these VTE-parameters is governed by the value of the corresponding "capability" VTE-parameter (see 17.1); the instances are treated as an ordered list and each occurrence may be referenced by its position in the list.

Table 6 -- Repertoire VTE-parameters

Parameter	Value
emphasis-assignment	each instance is an emphasis name (default = "normal" for the case where emphasis-capability = 1).
foreground-colour-assignment	each instance is a foreground-colour name (default = "device-dependent" for the case where foreground-colour-capability has the value 1).
background-colour-assignment	each instance is a background-colour name (default = "device-dependent" for the case where background-colour-capability has the value 1).

Where the corresponding "capability" VTE-parameter is greater than one, no default values are defined for the occurrences of this VTE-parameter.

Explicit values for emphasis-assignment are as defined for emphasis types in ISO 6429.

Explicit values for the foreground and background colour assignment VTE-parameters are as defined in ISO 6429.

NOTE -- For the default values "normal" and "device-dependent", the device may use whatever level of emphasis or colour it has available.

17.3 Tertiary VTE-Parameter

The font-assignment VTE-parameter is specified in table 7. Multiple instances of this tertiary VTE-parameter may be defined. Its occurrences are controlled by the repertoire-capability and font-capability VTE-parameters. The value of repertoire-capability implies a list of repertoires as defined in 17.1 and 17.2.2. With each of these repertoires is associated an instance of font-capability as defined in 17.2.2. Each such instance implies an ordered list of font-assignment VTE-parameter values, the fonts being identified by their position in the list. Not all the implied entries need to be given an explicit assignment. The lists themselves may be considered as being in an ordered list of the same ordering as the repertoire assignments. Both individual list entries and complete lists may be replaced by a "null" value to act as a placeholder when it is not required to give an explicit value.

Table 7 -- Font-assignment VTE-parameter

Parameter	Value
font-assignment	each instance is a font name (default = "device-dependent" for the case where font-capability has the value 1).

Where font-capability is greater than one, no default values are defined for the implied occurrences of this VTE-parameter.

NOTE -- For the default value "device-dependent", the device may use whatever font it has available, see C.14.

18 OPERATIONS ON DISPLAY OBJECTS

18.1 Addressing Operations

18.1.1 Implicit Addressing

When a value is written to the primary attribute of an array element, the x-coordinate value of the display pointer is automatically incremented by one. This mechanism is automatically disabled when the x-coordinate value reaches a value one greater than the x-dimension bound. No text or attribute operation may be initiated with the display pointer in this position. This mechanism may be re-enabled by explicit adjustment of the display pointer x-coordinate to a lower value. The mechanism is never disabled if the x-dimension is unbounded.

18.1.2 Explicit Addressing

The following operations are provided for the explicit modification of the display pointer position:

- a) primitive operations in which not all of the defined dimensions need to be included in any one invocation; any not included remain unchanged:

- 1) pointer absolute k,m,n (x:=k; y:=m; z:=n);
- 2) pointer signed (x:=x+p; y:=y+q; z:=z+r);
 relative p,q,r
- 3) home (x:=1; y:=1; z:=1).

- b) macro operations which may only be used in the form stated:

- 1) next X-array (y:=y+1; x:=Xmin);
- 2) previous X-array (y:=y-1; x:=Xmin);
- 3) next Y-array (z:=z+1; y:=Ymin; x:=Xmin);
- 4) previous Y-array (z:=z-1; y:=Ymin; x:=Xmin);

where Xmin, Ymin are the lower update-window bounds for the destination X-array and Y-array respectively.

The availability of these operations is determined by the addressing VTE-parameters negotiated for the current-VTE. The use of these operations does not affect the value of the

content of any previously existing array element nor does it generate any new array elements. Updating an array element does not automatically initiate any of these explicit addressing operations.

Any explicit addressing operation is invalid if the resulting pointer value would be, in any dimension, above the bound of a bounded dimension or below the origin (coordinate value 1) of any dimension.

NOTE -- The left-to-right order for the macro operations is necessary to ensure correct operation as Xmin is a function of y and z, and Ymin is a function of z (see 18.1.3).

18.1.3 Update-window Mechanism

The update-window mechanism enables VT-users to impose update addressing constraints which supplement those provided by array bounds. It places lower limits on locations which may be updated by text, attribute or erase operations on a display object. It does not, however, constrain the upper limit of text operations.

The use of the update-window mechanism may be negotiated independently for each defined array dimension (see C.3).

The update-window-size may have values defined such that, for any bounded dimension

$$1 \leq \text{update-window-size} \leq \text{array-bound}$$

where array-bound = x-bound, y-bound or z-bound as appropriate; for an unbounded dimension no upper limit on update-window-size applies.

In a dimension for which an update-window is defined, the lower bound of the update-window is denoted as coordinate value C where

$$C = \text{MAX}(B - \text{update-window-size} + 1, 1)$$

where B (the upper bound of the update-window) is the highest coordinate value at which an update by a text operation has been made. An array element may be updated by a text operation if its address is $\geq C$. A text operation update at an address $> B$ causes an instant move of the lower and upper update-window bounds.

The update-window does not constrain the position of the display pointer.