



INTERNATIONAL TELECOMMUNICATION UNION

CCITT

THE INTERNATIONAL
TELEGRAPH AND TELEPHONE
CONSULTATIVE COMMITTEE

T.433

(11/1988)

SERIES T: TERMINAL EQUIPMENT AND PROTOCOLS
FOR TELEMATIC SERVICES

**DOCUMENT TRANSFER AND MANIPULATION
(DTAM) – SERVICES AND PROTOCOLS –
PROTOCOL SPECIFICATION**

Reedition of CCITT Recommendation T.433 published in
the Blue Book, Fascicle VII.7 (1988)

NOTES

- 1 CCITT Recommendation T.433 was published in Fascicle VII.7 of the *Blue Book*. This file is an extract from the *Blue Book*. While the presentation and layout of the text might be slightly different from the *Blue Book* version, the contents of the file are identical to the *Blue Book* version and copyright conditions remain unchanged (see below).
- 2 In this Recommendation, the expression “Administration” is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

**DOCUMENT TRANSFER AND MANIPULATION (DTAM) –
SERVICES AND PROTOCOLS – PROTOCOL SPECIFICATION**

0 Introduction

This Recommendation specifies the protocol for the services provided by an application-service-element, the Document Transfer and Manipulation Service Element (DTAM) to support applications in a distributed telematic systems environment. This Recommendation is one of a set of Recommendations specifying the protocols for sets of application-service-elements specifically used by a number of applications.

1 Scope and field of application

This Recommendation specifies the protocol and procedures for the Document Transfer and Manipulation Service Element (DTAM). The DTAM services are provided in conjunction with the Association Control Service Element (ACSE) service (Recommendation X.217), and the Presentation-service (Recommendation X.216) or the Session-service (Recommendation X.215). Depending on the mapping, Recommendation T.62 bis may also apply.

The DTAM procedures are defined in terms of:

- a) the interaction between peer DTAM protocol machines through the use of the ACSE-service and Presentation-service or Session-service; and
- b) the interactions between the DTAM protocol machine and its service-user.

This Recommendation specifies conformance requirements for systems implementing these procedures.

The use of RTSE and/or ROSE is for further study.

2 References

References are listed in Recommendation T.432.

3 Definitions and abbreviations

Terms and abbreviations are defined in Recommendation T.431. The definitions of service primitive names given in Recommendation T.432 are used in this Recommendation.

4 Conventions

This Recommendation specifies the APDU Fields. In 6, tables are presented for each DTAM APDU. Each field is summarized by the following notation:

- M presence is mandatory
- U presence is a DTAM service-user option
- req source is related request primitive
- ind sink is related indication primitive
- rsp source is related response primitive
- cnf sink is related confirm primitive
- sp source or sink is the DTAM-PM

The structure of each DTAM APDU is specified in 8 using the abstract syntax notation of Recommendation X.208.

5 Overview of the protocol

5.1 Service provision

The protocol specified in this Recommendation provides the DTAM services defined in Recommendation T.432. These services are listed in Table 1/T.433.

5.2 Relationship with other ASEs and lower layer services

5.2.1 ACSE service (when RTSE is not used)

The DTAM services require access to the A-ASSOCIATE, A-RELEASE, A-ABORT, and A-P-ABORT services. The inclusion of the DTAM in an application-context precludes the use of any of the above ACSE services by any other ASE or the use-element.

The Transparent Mode of DTAM implies that ACSE can pass through it.

TABLE 1/T.433
DTAM services summary

Service	Type
D-INITIATE	confirmed
D-TERMINATE	confirmed
D-P-ABORT	provider-initiated
D-U-ABORT	unconfirmed
D-CAPABILITY	confirmed
D-TRANSFER	provider-confirmed
D-TYPED-DATA	unconfirmed
D-CREATE	unconfirmed
D-DELETE	unconfirmed
D-MODIFY	unconfirmed
D-CALL	unconfirmed
D-REBUILD	unconfirmed
D-TOKEN-GIVE	unconfirmed
D-CONTROL-GIVE	unconfirmed
D-TOKEN-PLEASE	unconfirmed
D-P-EXCEPTION-REPORT	provider-initiated
D-U-EXCEPTION-REPORT	unconfirmed

Note – D-REBUILD service is for further study.

5.2.2 RTSE service

The use of this ASE is for further study.

5.2.3 ROSE service

The use of this ASE is for further study.

5.2.4 Presentation-service

DTAM services may require access to the P-ACTIVITY-START, P-DATA, P-MINOR- SYNCHRONIZE, P-ACTIVITY-END, P-ACTIVITY-INTERRUPT, P-ACTIVITY-DISCARD, P-U-EXCEPTION-REPORT, P-ACTIVITY-RESUME, P-P-EXCEPTION-REPORT, P-TOKEN-PLEASE, P-CONTROL-GIVE and P-TOKEN-GIVE services. This Recommendation recognizes that the ACSE services require access to the P-CONNECT, P-RELEASE, P-U-ABORT and P-P-ABORT services. The inclusion of the DTAM in an application-context precludes the use of any of the above, or of any other, presentation-services by any other ASE or the user element.

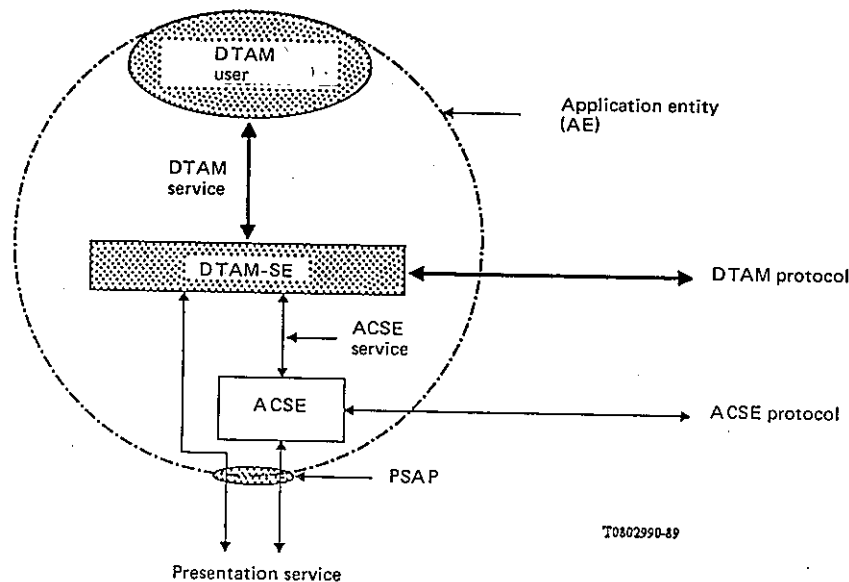
5.2.5 Recommendation X.215 session-service

In the Transparent Mode of operation APDUs defined in DTAM are directly mapped to the session service defined in Recommendation X.215. When the Transparent Mode is used the procedure described in Recommendation T.62 bis also apply.

DTAM services may require access to the S-CONNECT, S-ACTIVITY-START, S-DATA, S-MINOR- SYNCHRONIZE, S-ACTIVITY-END, S-ACTIVITY-INTERRUPT, S-ACTIVITY-DISCARD, S-U-EXCEPTION-REPORT, S-ACTIVITY-RESUME, S-P-EXCEPTION-REPORT, S-TOKEN-PLEASE, S-CONTROL-GIVE, P-TOKEN-GIVE, S-RELEASE, S-U-ABORT and S-P-ABORT services.

5.3 Model of telematic protocol architecture (TPA)

The DTAM operates between two DTAM Protocol Machines (DTAM-PMs) in the Application layer of the OSI model. Protocol elements are exchanged between DTAM-PMs, using the Session service as defined in Recommendation X.215 or the services of ACSE and of the Presentation Layer as defined in Recommendations X.217 and X.216 respectively. The model for Telematic Protocol Architecture (TPA) is illustrated in Figure 1/T.433. This application layer protocol architecture is composed of the ACSE (Association Control Service Element), DTAM-SE (Service Element), and DTAM users. Use of the Reliable Transfer Service Element (RTSE), Remote Operation Service Element (ROSE) and Message Handling Systems (MHS) is for further study.



Note – In the case of use of the Session-service (Transparent Mode), the appropriate DTAM APDUs are directly mapped to the Session-service primitives.

FIGURE 1/T.433

**Telematic protocol architecture (TPA)
model in application layer**

5.3.1 *Functions of DTAM user*

DTAM users have the role of accurately reflecting the actual telematic user (i.e. terminal or system user) intentions in communication, and have functions to perform the applications (Document bulk transfer, document manipulation, document transfer and manipulation etc.) on behalf of the actual user. This mechanism is provided by the use of the DTAM-SE through the DTAM service defined in Recommendation T.432. The DTAM service is the logical interface between the DTAM user and DTAM service-provider for data handling, and is independent of specific hardware and software technique.

The DTAM user as an Application Service Element (including the user element) may be capable of interpreting the meaning of the content of an exchanged document. For example, the retrieval command carried during information retrieval is not interpreted by the DTAM, but by the DTAM user.

5.3.2 *Functions of DTAM service-provider*

To realize single-source management of document architecture for telematic services, DTAM service-provider provides the following communication functions.

1) *Association use control (kernel)*

DTAM provides the trigger for the use of the association given in ACSE, and controls association use during communication (termination, abort, etc.). Applying the Session-service to the lower layer functions of DTAM, this association use control will be mapped directly onto the session kernel functional unit.

2) *DTAM capability*

The DTAM capability is defined by a set of parameters in order to specify the communication features which contains the parameters:

- a) document application profile;
- b) operational application profile;
- c) non-basic document characteristics;
- d) non-basic structural characteristics, etc.

3) *Data transmission function*

DTAM provides functions for document bulk transfer, document manipulations and typed data transmission as follows:

a) *Document bulk transfer*

DTAM provides a function to transmit the document in bulk under the communications environment negotiated by D-INITIATE service and additionally by D-CAPABILITY service;

b) *Document manipulations*

DTAM provides a function partially modifying a document seen by both users, by generating, revising or deleting structures (pages, blocks etc.) of an existing document or to create a new document by generating structure of ODA and Operational Structure;

c) *Typed data transmission*

DTAM optionally provides a typed data transmission function which is independent of data token control.

4) *Document remote access*

For further study.

5) *Document remote management*

For further study.

6) *Token control*

DTAM optionally provides the function of Token control to handle the data token for dialogue.

7) *Reliable transfer (support function)*

DTAM optionally provides the function of reliable transfer to ensure reliable communication. Two Reliable Transfer Modes are introduced (see § 6.6.1.4).

8) *Exception report*

DTAM optionally provides the exception reporting function for error control during the DTAM communication.

9) *Storage capacity negotiation*

DTAM optionally provides the Storage Capacity Negotiation to indicate its own capacity to the peer.

6 Elements of procedure

This section identifies all the types of protocol data units which constitute the elements of the DTAM protocol between two DTAM-protocol-machines (DTAM-PMs). A protocol data unit (PDU) is the smallest quantity of information exchanged between DTAM-PMs which has a self-contained semantic significance.

When a DTAM service primitive is received from the DTAM user, DTAM transmits the DTAM primitive data to the opposite DTAM through the DTAM protocol, then the opposite DTAM generates the DTAM service primitives and notifies its DTAM user. The DTAM protocol data units (D-PDU) are shown in Table 2/T.433.

Individual parameters of DTAM service primitives are, in principle, all mapped to individual PDU parameters, but there are PDU including parameters other than those specified in service primitives, such as those generated by DTAM itself. For example, D-INITIATE-REQ PDU also includes the DTAM protocol version parameter, which is used to negotiate the version of protocol between the DTAM-PMs. Note that the DTAM user is not concerned with this DTAM negotiation.

The PDUs are here identified symbolically with minimal reference to their mapping on to the lower layer service functions which implement them, thus no differentiation is made, in this section, between PDUs which are effected as specific Presentation service primitives and PDUs which are transferred as DTAM PDUs using the Presentation service data transfer functions. Details of PDU mapping and encoding are given in § 8.

PDUs are given both full names, which should be used outside the context of this Recommendation, and abbreviated names which are used within this Recommendation for brevity. The full names consist of one or two words descriptive for the purpose of the PDU, prefixed by D- and, in the case of request/response pairs of PDUs, suffixed by -REQ or -RESP as appropriate. The abbreviated names are three letters each, with Q or R appended in the case of request/response pairs.

6.1 *Summary list of DTAM protocol data units*

TABLE 2/T.433

DTAM protocol data units

Functional Units	PDU abbrev.	Protocol elements (PDU)	Reference
Association use control (kernel)	DINQ	D-INITIATE-REQ	6.2
	DINR	D-INITIATE-RESP	6.2
	DTEQ	D-TERMINATE-REQ	6.3
	DTER	D-TERMINATE-RESP	6.3
	DAB	D-ABORT	6.4
Capability	DCPQ	D-CAPABILITY-REQ	6.5
	DCPR	D-CAPABILITY-RESP	6.5
Document Bulk transfer	None	None	6.6

TABLE 2/T.433 (Cont.)

Functional Units	PDU abbrev.	Protocol elements (PDU)	Reference
Document unconfirmed manipulation	DCR	D-CREATE	6.7
	DDL	D-DELETE	6.7
	DMD	D-MODIFY	6.7
	DCL	D-CALL	6.7
	DRD	D-REBUILD [Further study]	6.7
Document confirmed manipulation		[Further study]	6.8
Typed data transmission	DTD	D-TYPED-DATA	6.9
Remote document access		[Further study]	6.10
Remote document management		[Further study]	6.11
Token control	DTP	D-TOKEN-PLEASE	6.12
Exception report	None	None - user-exception-report - provider-exception-report	6.13
Reliable transfer (support)	None	None	6.6

6.2 DTAM association establishment

6.2.1 Purpose

The DTAM association establishment procedure is used to establish an association of DTAM between two AEs. It supports the D-INITIATE service.

6.2.2 APDUs used

The DTAM association establishment procedure uses the D-INITIATE-REQ (DINQ) and the D-INITIATE-RESP (DINR) APDUs.

6.2.2.1 DINQ APDU

The fields of the DINQ APDU are listed in Table 3/T.433.

TABLE 3/T.433
DINQ APDU fields

Field name	Presence	Source	Sink
Service classes (see Note 1)	(see Note 2)	request	indication
Telematic requirements (see Note 1)	M	request	indication
Application capabilities	M	request	indication
Protocol version (see Note 1)	U	sp	sp
DTAM QOS (see Note 1)	U	request	indication
Account (see Note 1)	U	request	indication
Window size	U	request	indication
Storage capacity	U	request	indication
User information (see Note 1)	U	request	indication

Note 1 – These parameters are not applicable in transparent mode.

Note 2 – The use of this parameter is for further study.

6.2.2.2 DINR APDU

The fields of the DINR APDU are listed in Table 4/T.433.

TABLE 4/T.433
DINR APDU fields

Field name	Presence	Source	Sink
Telematic requirements (see Note)	U	response	confirmation
Application capabilities	M	response	confirmation
Protocol version (see Note)	U	sp	sp
DTAM QOS (see Note)	U	response	confirmation
Result (see Note)	M	response	confirmation
Window size	U	response	confirmation
Storage capacity	U	response	confirmation
User information (see Note)	U	response	confirmation

Note – These parameters are not applicable in transparent mode.

6.2.3 *DTAM association establishment procedure*

6.2.3.1 *DTAM association establishment procedure mapped onto ACSE service (normal mode: OSI)*

This procedure is driven by the following events:

- a) a D-INITIATE request primitive from the requestor;
- b) a DINQ APDU as User Data on an A-ASSOCIATE indication primitive;
- c) a D-INITIATE response primitive from the responder; and
- d) an A-ASSOCIATE confirm primitive (that may contain a DINR APDU);

6.2.3.1.1 *D-INITIATE request primitive*

6.2.3.1.1.1 The requesting DTAM-PM forms a DINQ APDU from parameter values of the D-INITIATE request primitive and its stored data in DTAM-PM (the Protocol Version field, etc.) It issues an A-ASSOCIATE request primitive also using information from the D-INITIATE request primitive. The User Data parameter of the A-ASSOCIATE request primitive contains the DINQ APDU.

6.2.3.1.1.2 The requesting DTAM-PM waits for a primitive from the ACSE service-provider, and does not accept any other primitive from the requestor other than a D-U-ABORT request primitive.

6.2.3.1.2 *DINQ APDU*

6.2.3.1.2.1 The responding DTAM-PM receives a DINQ APDU from its peer as User data on an A-ASSOCIATE indication primitive. If any of the parameters of the A-ASSOCIATE indication primitive or the fields in the DINQ APDU are unacceptable to this DTAM-PM, it forms a DINR APDU with the appropriate rejecting Result field, and sends the DINR APDU as User Data on an A-ASSOCIATE response primitive. The Result parameter on the A-ASSOCIATE response primitive specifies "user-rejection". The DTAM-PM does not issue a D-INITIATE indication primitive to the responder, and the association is not established.

6.2.3.1.2.2 If the A-ASSOCIATE indication primitive and its DINQ APDU are acceptable to the responding DTAM-PM, it issues a D-INITIATE indication primitive to the responder. The D-INITIATE indication primitive parameters are derived from the DINQ APDU and from the A-ASSOCIATE indication primitive. The DTAM-PM waits for a D-INITIATE response primitive from the responder and does not accept any other primitives from the responder except for the D-U-ABORT request primitive.

6.2.3.1.3 *D-INITIATE response primitive*

6.2.3.1.3.1 When the DTAM-PM receives the D-INITIATE response primitive, the Result parameter specifies whether the responder has accepted or rejected the DTAM association. The DTAM-PM forms a DINR APDU using the D-INITIATE response primitive parameters. The DINR APDU is sent as the User Data parameter on the A-ASSOCIATE response primitive.

6.2.3.1.3.2 If the responder accepted the DTAM association request, the Result parameter on the related A-ASSOCIATE response primitive specifies "accepted", and the Result field of the outgoing DINR APDU also specifies "accepted". The DTAM association is established.

6.2.3.1.3.3 If the responder rejected the DTAM association request, the Result parameter on the related A-ASSOCIATE response primitive specifies "Result: rejected (permanent or transient)" and "Source: ACSE service-user", and the Result field of the outgoing DINR APDU contains the appropriate rejection value. The DTAM association is not established.

6.2.3.1.4 *A-ASSOCIATE confirm primitive*

6.2.3.1.4.1 The requesting DTAM-PM receives an A-ASSOCIATE confirm primitive. The following situations are possible:

- a) the DTAM association has been accepted;
- b) the responding DTAM-PM or the responder has rejected the DTAM association; or
- c) the association service-provider has rejected the related association.

6.2.3.1.4.2 If the DTAM association was accepted, the A-ASSOCIATE confirm primitive Result parameter specifies "accepted". The User Data parameter contains a DINR APDU, and the Result field of the DINR APDU also specifies "accepted". The requesting DTAM-PM issues a D-INITIATE confirm primitive to the requestor based on parameters from the A-ASSOCIATE confirm primitive and from the DINR APDU. The D-INITIATE confirm primitive Result parameter specifies "accepted", and the DTAM association is established.

6.2.3.1.4.3 If the DTAM association was rejected by either the responding DTAM-PM or by the responder, the A-ASSOCIATE confirm primitive Result parameter specifies "Result: rejected (permanent or transient)" and "Source: ACSE service-user". The User Data parameter contains a DINR APDU, and the Result field of the DINR APDU indicates the reason for rejection. The requesting DTAM-PM issues a D-INITIATE confirm primitive to the requestor based on parameters from the A-ASSOCIATE confirm primitive and from the DINR APDU. The D-INITIATE confirm primitive Result parameter contains the appropriate rejection value. The DTAM association is not established.

6.2.3.1.4.4 If the association was rejected by the association service-provider, the A-ASSOCIATE confirm primitive Result parameter specifies "Result: rejected (permanent or transient)" and "Source: ACSEservice-provider". In this situation, the User Data field is not used by the requesting DTAM-PM. The requesting DTAM-PM issues a D-INITIATE confirm primitive with the appropriate Result parameter. The DTAM association is not established.

6.2.3.2 *DTAM association establishment procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

- a) a D-INITIATE request primitive from the requestor;
- b) a DING APDU as User Data on an S-CONNECT indication primitive;
- c) a D-INITIATE response primitive from the responder; and
- d) an S-CONNECT confirm primitive (that may contain a DINR APDU);

6.2.3.2.1 *D-INITIATE request primitive*

6.2.3.2.1.1 The requesting DTAM-PM forms a DING APDU from parameter values of the D-INITIATE request primitive and its stored data in DTAM-PM (the Checkpoint Window field, etc.). It issues an S-CONNECT request primitive also using information from the D-INITIATE request primitive. The User Data parameter of the CONNECT request primitive contains the DING APDU.

6.2.3.2.1.2 The requesting DTAM-PM waits for a primitive from the Session service-provider and does not accept any other primitive from the requestor other than a D-U-ABORT request primitive.

6.2.3.2.2 *DING APDU*

6.2.3.2.2.1 The responding DTAM-PM receives a DING APDU from its peer as User Data on an S-CONNECT indication primitive. If any of the parameters of the S-CONNECT indication primitive or the fields in the DING APDU are unacceptable to this DTAM-PM (e.g. no Session User Data in the S-CONNECT indication,) it issues an S-CONNECT response primitive specified "ss-user-rejection". In this situation, the responding session service-provider issues RSSN (Response Session Start Negative). The DTAM-PM does not issue a D-INITIATE indication primitive to the responder. The association is not established.

6.2.3.2.2.2 If the S-CONNECT indication primitive and its DING APDU are acceptable to the responding DTAM-PM, it issues a D-INITIATE indication primitive to the responder. The D-INITIATE indication primitive parameters are derived from the DING APDU. The DTAM-PM waits for a D-INITIATE response primitive from the responder and does not accept any other primitives from the responder except for the D-U-ABORT request primitive.

6.2.3.2.3 *D-INITIATE response primitive*

6.2.3.2.3.1 When the DTAM-PM receives the D-INITIATE response primitive, the Result parameter specifies whether the responder has accepted or rejected the DTAM association. If the DTAM association is accepted, the DTAM-PM forms a DINR APDU using the D-INITIATE response primitive parameters. The DINR APDU is sent as the User Data parameter on the S-CONNECT response primitive.

6.2.3.2.3.2 If the responder accepted the DTAM association request, the Result parameter on the related S-CONNECT response primitive specifies "accept". The DTAM association is established.

6.2.3.2.3.3 If the responder rejected the DTAM association request, the Result parameter on the related S-CONNECT response primitive specifies "user-rejection" and DTAM-PM does not send DINR APDU.

6.2.3.2.4 *S-CONNECT confirm primitive*

6.2.3.2.4.1 The requesting DTAM-PM receives an S-CONNECT confirm primitive. The following situations are possible;

- a) the DTAM association has been accepted;
- b) the responding DTAM-PM or the responder has rejected the DTAM association; or
- c) the Session service-provider has rejected the related association.

6.2.3.2.4.2 If the DTAM association was accepted, the S-CONNECTION confirm primitive Result parameter specifies "accept". The User Data parameter contains a DINR APDU. The requesting DTAM-PM issues a D-INITIATE confirm primitive to the requestor based on parameters from the S-CONNECT confirm primitive and from the DINR APDU. The D-INITIATE confirm primitive Result parameter specifies "accepted". The DTAM association is established.

6.2.3.2.4.3 If the DTAM association was rejected by either the responding DTAM-PM or by the responder, the S-CONNECT confirm primitive Result parameter specifies "user-rejection" and there is no User Data (DINR APDU) in this confirm primitive. The requesting DTAM-PM issues a D-INITIATE confirm primitive to the requestor based on parameters from the S-CONNECT confirm primitive. The D-INITIATE confirm primitive Result parameter contains the value of "user-rejection", and the DTAM association is not established.

6.2.3.2.4.4 If the association was rejected by the session service-provider, the S-CONNECT confirm primitive Result parameter specifies "provider-rejection". In this situation, the User Data field is not used by the requesting DTAM-PM. The requesting DTAM-PM issues a D-INITIATE confirm primitive with the appropriate Result parameter. The DTAM association is not established.

6.2.4 *Use of the DINQ/DINR APDU fields*

The DINQ APDU and DINR APDU fields are used as follows.

6.2.4.1 *Service classes*

The use of this parameter is for further study.

6.2.4.2 *Telematic requirements*

This is the Telematic Requirements parameter value from the D-INITIATE request/response primitives. It appears as the Telematic Requirements parameter value of D-INITIATE indication/confirm primitives respectively. If the Telematic Requirements proposed by the requestor are not acceptable to the responder, the DTAM association fails to be established.

6.2.4.3 *Application capabilities*

This is the Application Capabilities parameter value from the D-INITIATE request/response primitives. It appears as the Application Capabilities parameter value of the D-INITIATE indication/confirm primitives respectively. This parameter consists of sets of the following sub- parameters.

6.2.4.3.1 *Document application profile*

The value of this parameter is either an Octet String or ASN.1 object identifiers. The Octet String designates the document application profile in line with the Recommendation T.73 (Document Application Profile – T.73). The ASN.1 object identifier must conform to the rules specified in ISO 8824 and designate an application profile defined in accordance with the rules specified in Recommendation T.411 (Document Application Profiles).

6.2.4.3.2 *Document architecture class*

The value of this parameter is "formatted".

6.2.4.3.3 *Non basic document characteristics*

The value of this parameter is any combination of Non Basic Document Characteristics defined in Recommendation T.414.

6.2.4.3.4 *Non basic structural characteristics*

The value of this parameter is any combination of Non Basic Structural Characteristics defined in Recommendation T.414.

6.2.4.3.5 *Operational application profile*

The detailed specification of the Operational Application Profile is for further study.

6.2.4.4 *Protocol version*

This identifies the version of the DTAM protocol in use by the requesting DTAM-PM.

6.2.4.5 *DTAM QOS*

DTAM QOS is left for further study.

6.2.4.6 *Account*

The account parameter identifies the account to which costs incurred in the DTAM association which is being established are to be charged.

Note – The use of this parameter is for further study.

6.2.4.7 *Window size*

The requested checkpoint window parameter indicates, for each direction of transmission, the maximum number of checkpoints which may remain unacknowledged. This parameter is conditional upon the recovery or restart procedures under the reliable transfer, in which case it is mandatory. Checkpoints are only inserted by the sender of a document. Values of this parameter may be the reason for subsequent termination. The continued progress of the service is only guaranteed if the entity acting as receiver gives acknowledgments within this limit. The window size is stated independently by each entity as the maximum value for when that entity is the receiving entity. There is no negotiation. The values for each direction of data transfer are not necessarily the same. The parameter is an integer.

6.2.4.8 *Storage capacity*

In a Normal Mode, this parameter is optionally used by each of two DTAM-PMs to indicate its own capacity to the peer. After the negotiation, if the storage capacity of the receiving DTAM-PM is smaller than the largest segment of document information (see 6.6) according to the checkpoint rule, the sending DTAM-PM shall not transfer the document and D-P-EXCEPTION indication should be issued to the requesting DTAM user (sender of documents).

However, for some applications under a Transparent Mode, this parameter is used by the sending DTAM-PM to indicate a 'required storage capacity' to the peer machine. The receiving DTAM-PM uses this parameter to respond whether it is able to provide this storage capacity or not, so as to maintain compatibility with the old implementation based on Recommendation T.73.

6.2.4.9 *Result*

If the DINQ APDU was rejected by the responding DTAM-PM (i.e., a D-INITIATE indication primitive was not issued to the responder), this field is supplied by the responding DTAM-PM, otherwise, this field is the Result parameter from the D-INITIATE response primitive. In either situation, it appears as the Result parameter on the D-INITIATE RESP (DINR) APDU. This field can take one of the following symbolic values:

- accepted;
- rejected by responder (reason-not-specified);
- rejected by responder (protocol Version-not-supported);
- rejected by responder (DTAMQOS-not-supported);
- rejected by responder (application-context-not-supported);
- rejected by responding DTAM-PM.

6.2.4.10 *User information*

This is the User Information parameter from the D-INITIATE request and response primitive. It appears as the User Information parameter of the D-INITIATE indication and confirm primitive respectively, if issued.

6.2.5 *Collisions and interactions*

For further study.

6.3 *Normal termination of a DTAM association*

6.3.1 *Purpose*

This procedure is used for the normal termination of a DTAM association by an AE without loss of information in transit. It supports the D-TERMINATE service.

6.3.2 *APDUs used*

The normal termination procedure uses the D-TERMINATE-REQ (DTEQ) APDU and the D-TERMINATE-RESP (DTER) APDU.

6.3.2.1 DTEQ APDU

The fields of the DTEQ APDU are listed in Table 5/T.433.

TABLE 5/T.433
DTEQ APDU fields

Field name	Presence	Source	Sink
User information (see Note)	U	request	indication

Note – This parameter is not applicable in transparent mode.

6.3.2.2 DTER APDU

The fields of the DTER APDU are listed in Table 6/T.433.

TABLE 6/T.433
DTER APDU fields

Field name	Presence	Source	Sink
Charging (see Note)	U	response	confirmation
User information (see Note)	U	response	confirmation

Note – These parameters are not applicable in transparent mode.

6.3.3 Normal termination procedure

6.3.3.1 Normal termination procedure mapped onto ACSE service (normal mode)

This procedure is driven by the following events:

- a D-TERMINATE request primitive from the requestor;
- a DTEQ APDU as User Data on an A-RELEASE indication primitive;
- a D-TERMINATE response primitive from the responder; and
- a DTER APDU as User Data on an A-RELEASE confirm primitive.

6.3.3.1.1 D-TERMINATE request primitive

6.3.3.1.1.1 When a D-TERMINATE request primitive is received, the DTAM-PM sends a DTEQ APDU as User Data on an A-RELEASE request primitive using the parameters from the D-TERMINATE request primitive.

Note – The requestor is required to meet the association (presentation and session) requirements in order to issue a D-TERMINATE request primitive.

6.3.3.1.1.2 The requesting DTAM-PM now waits for a primitive from the association service-provider. It does not accept any primitives from the requestor other than a D-U-ABORT request primitive.

6.3.3.1.2 DTEQ APDU

6.3.3.1.2.1 When the responding DTAM-PM receives the DTEQ APDU as User Data on an A-RELEASE indication primitive, it issues a D-TERMINATE indication primitive to the responder.

6.3.3.1.3 *D-TERMINATE response primitive*

6.3.3.1.3.1 The responding DTAM-PM forms a DTER APDU from the response primitive parameters. The DTER APDU is sent as User Data on an A-RELEASE response primitive. The Result parameter of A-RELEASE response has the value "affirmative".

Note – The responder is able to reject the termination request of DTAM association only in the case of selecting a negotiated release session functional unit. The use of this functional unit is for further study.

6.3.3.1.4 *DTER APDU*

6.3.3.1.4.1 The requesting DTAM-PM receives an A-RELEASE confirm primitive containing a DTER APDU from its peer. The Result parameter on the A-RELEASE confirm specifies either that the responder agrees or disagrees that the DTAM association may be terminated. The requesting DTAM-PM forms a D-TERMINATE confirm primitive from the DTER APDU.

6.3.3.2 *Normal termination procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

- a) a D-TERMINATE request primitive from the requestor;
- b) an S-RELEASE indication primitive without sending DTEQ APDU;
- c) a D-TERMINATE response primitive from the responder; and
- d) an S-RELEASE confirm primitive without sending DTER APDU.

6.3.3.2.1 *D-TERMINATE request primitive*

6.3.3.2.1.1 When a D-TERMINATE request primitive is received, the DTAM-PM issues an S-RELEASE request primitive without any SS-user-data.

Note – The requestor is required to meet the association (presentation and session) requirements in order to issue a D-TERMINATE request primitive.

6.3.3.2.1.2 The requesting DTAM-PM now waits for a primitive from the Session service-provider. It does not accept any primitives from the requestor other than a D-U-ABORT request primitive.

6.3.3.2.2 *Implicit DTEQ APDU*

6.3.3.2.2.1 When the responding DTAM-PM receives an S-RELEASE indication primitive, it issues a D-TERMINATE indication primitive to the responder without any parameters.

6.3.3.2.3 *D-TERMINATE response primitive*

6.3.3.2.3.1 The responding DTAM-PM forms an S-RELEASE response from the D-TERMINATE response primitive parameters. The Result parameter of S-RELEASE response has the value "affirmative".

6.3.3.2.4 *Implicit DTER APDU*

6.3.3.2.4.1 The requesting DTAM-PM receives an S-RELEASE confirm primitive containing no DTAM APDU from its peer. The Result parameter on the S-RELEASE confirm always specifies "affirmative". The requesting DTAM-PM forms a D-TERMINATE confirm primitive from the S-RELEASE confirm primitive and issues it to the requestor with no parameters.

6.3.4 *Use of the DTEQ APDU fields*

The DTEQ APDU fields are used as specified below.

6.3.4.1 *User information*

This is the User Information parameter on the D-TERMINATE request primitive. It appears as the User Information parameter of the D-TERMINATE indication primitive.

6.3.5 *Use of the DTER APDU fields*

The DTER APDU fields are used as specified below.

6.3.5.1 *Charging*

The charging parameter conveys information on the costs attributed to the account during the DTAM association which is being released. The value of this parameter is for further study. The charging parameter if present at the end of a DTAM association, only if the account parameter was present at the beginning of that DTAM association. It is not mandatory to return a charge if that charge is zero.

6.3.5.2 *User information*

This is the User Information parameter from the D-TERMINATE response primitive. It appears as the User Information parameter on the D-TERMINATE confirm primitive.

6.3.6 *Collisions and interactions*

6.3.6.1 *D-TERMINATE service*

Overlapping attempts by request in both AEs to terminate their DTAM association are governed by the A-RELEASE service or S-RELEASE Session service. The DTAM association is terminated.

Note – A D-terminate service collision can not occur if session tokens were selected for the association. Only a request in the AE that owns all of the available session tokens can issue the D-TERMINATE request primitive.

6.3.6.2 *D-U-ABORT service, DAB APDU or A-P-ABORT service*

If either DTAM-PM receives a D-U-ABORT request primitive, a DAB APDU (as User Data on a A(or S)-U-ABORT indication primitive) or a A(orS)-P-ABORT indication primitive, it discontinues the normal DTAM association termination procedure, and instead follows abnormal termination procedure.

6.4 *Abnormal termination of a DTAM association*

6.4.1 *Purpose*

6.4.1.1 *The abnormal termination can be used at any time to force the abrupt termination of the DTAM association by a requestor in either DTAM user, by either DTAM-PM, by the ACSE service-provider or by the Session service-provider. It supports the D-U-ABORT, D-P-ABORT and A-P-ABORT or S-P-ABORT services.*

6.4.1.2 *The Abnormal Termination provides the following three procedures:*

- a) user-abort procedure;
- b) association-provider-abort procedure;
- c) transfer-abort procedure.

6.4.2 *APDUs used*

The abnormal termination uses the D-ABORT (DAB) APDU.

6.4.2.1 *DAB APDU*

The fields of the DAB APDU are listed in Table 7/T.433.

TABLE 7/T.433
DAB APDU fields

Field name	Presence	Source	Sink
Abort source (see Note)	M	sp	indication
Abort reason (see Note)	U	sp	indication
Reflect parameter (see Note)	U	sp	indication
User information (see Note)	U	request	indication

Note – These parameters are not applicable in transparent mode.

6.4.3 Abnormal termination procedure

6.4.3.1 Abnormal termination procedure mapped onto ACSE service (normal mode)

This procedure is driven by the following events:

User-abort procedure

- a D-U-ABORT request primitive from the requestor;
- a DAB APDU as User Data on an A-U-ABORT indication primitive;

Association-provider-abort procedure

- an A-P-ABORT indication primitive from the ACSE-service or

Transfer-abort procedure

- a severe error detected by a DTAM-PM.

6.4.3.1.1 D-U-ABORT request primitive (user-abort procedure)

6.4.3.1.1.1 When a DTAM-PM receives a D-U-ABORT request primitive, it sends a D-ABORT (DAB) APDU as User Data on an A-U-ABORT request primitive. The DAB APDU "Abort Source" field is specified as a "requestor". If the User Information parameter was included on the D-U-ABORT request primitive, it is included in the DAB APDU. The DTAM association is terminated.

6.4.3.1.2 DAB APDU

6.4.3.1.2.1 When a DTAM-PM receives an A-U-ABORT indication primitive, the User Data parameter contains the DAB APDU. The DTAM-PM issues a D-U-ABORT indication primitive with the Abort Source field of the DAB APDU. If a User Information field was contained in the DAB APDU, it is included in the D-U-ABORT indication primitive. The DTAM association is terminated.

6.4.3.1.3 A-P-ABORT indication primitive (association-provider-abort procedure)

6.4.3.1.3.1 When a DTAM-PM receives an A-P-ABORT indication primitive, the DTAM-PM issues a D-P-ABORT indication primitive to the DTAM user. The DTAM association is terminated.

6.4.3.1.3.2 An association-provider-abort is indicated to both DTAM-PMs by an A-P-ABORT indication primitive and may occur at any time. After such an event, when the Reliable Transfer Mode 2 was selected, the association-initiating DTAM-PM starts the association-recovery procedure.

Note – The association-recovery procedure is for further study.

6.4.3.1.3.3 If the association-provider-abort procedure was performed during the transfer procedure the requesting DTAM-PM starts the transfer-resumption procedure after the association-recovery procedure is successfully completed. If the association-recovery procedure was not successfully completed the requesting DTAM-PM performs the transfer-error procedure and the provider-abort procedure.

6.4.3.1.4 *Error detections by a DTAM-PM (transfer-abort procedure)*

6.4.3.1.4.1 When a DTAM-PM detects severe error situations, it performs the transfer-abort procedure followed by issuing a D-P-ABORT indication primitive.

6.4.3.1.4.2 The transfer-abort procedure is performed to send a DAB APDU as User Data on an A-U-ABORT request primitive. The DAB APDU "Abort Source" field is specified as a "DTAM service-provider" and additional DAB APDU parameters are specified to inform a peer DTAM-PM of the situation of the severe error. Following the transfer-abort procedure, the DTAM-PM issues a D-P-ABORT indication primitive to its service-user.

6.4.3.1.4.3 The use of association-recovery procedure (see § 6.6.8) is for further study.

6.4.3.2 *Abnormal termination procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

User-abort procedure

- a D-U-ABORT request primitive from the requestor;
- an S-U-ABORT indication primitive without sending a DAB APDU;

Association-provider-abort procedure

- an S-P-ABORT indication primitive from the Session-service or

Transfer-abort procedure

- a protocol error detected by a DTAM-PM.

6.4.3.2.1 *D-U-ABORT request primitive (user-abort procedure)*

6.4.3.2.1.1 When a DTAM-PM receives a D-U-ABORT request primitive, it issues a S-U-ABORT request primitive without sending a DAB APDU. The using of S-U-ABORT service will be interpreted as "Local Terminal Error". The DTAM association is terminated.

6.4.3.2.2 *Implicit DAB APDU*

6.4.3.2.2.1 When a DTAM-PM receives an S-U-ABORT indication primitive, the DTAM-PM issues a D-U-ABORT indication primitive with the Abort Source field as "requestor". The DTAM association is terminated.

6.4.3.2.3 *S-P-ABORT indication primitive (association-provider-abort procedure)*

6.4.3.2.3.1 When a DTAM-PM receives an S-P-ABORT indication primitive, the DTAM-PM issues a D-U-ABORT indication primitive to the responder. The DTAM association is terminated.

6.4.3.2.4 *Protocol errors (transfer-abort procedure)*

6.4.3.2.4.1 When a DTAM-PM detects an invalid condition such as an unexpected APDU, it issues an S-U-ABORT request primitive without DAB APDU as the User Data. The DTAM-PM also issues a D-P-ABORT indication primitive to its service-user. The DTAM association is terminated.

6.4.4 *Use of the ABORT APDU fields*

The ABORT APDU fields are used as specified below.

6.4.4.1 *Abort source*

This is supplied by the requesting DTAM-PM. It is included in the resulting D-U(orP)-ABORT indication primitive. This field can take one of the following symbolic values:

- DTAM service-provider; or
- requestor.

6.4.4.2 *Abort reason*

This field may contain one of the following values:

- local-system-problem
- invalid-parameter the invalid parameters are specified in the Reflected- parameter field
- unrecognized-activity

- temporary-problem no attempt at association-recovery should be made for a period of time determined by a local rule
- protocol-error of the DTAM-PM
- permanent-error this value is used solely by the DTAM provider-abort procedure in normal-mode
- transfer-completed the responding DTAM-PM could not discard an already completed transfer

6.4.4.3 *Reflected-parameter*

The Reflected-Parameter field is a bit string that identifies which parameters are regarded as invalid parameters in the primitive received from the used service by the aborting DTAM-PM before the association-abort. The order of the bits in the bit string is the same as the order of the parameters in the tables of service parameters in Recommendations X.216 and X.217 (i.e. bit 1 represents the first parameter, etc).

6.4.4.4 *User information*

This is the information parameter from the D-U-ABORT request primitive. It appears as the User Information parameter on the D-U-ABORT indication primitive.

6.4.5 *Collisions and interactions*

The abnormal termination procedure may be used whenever a DTAM association is established, is in process of being established, or is being normally terminated. This procedure disrupts any other currently active procedure. An A-P-ABORT indication primitive can disrupt the D-U-ABORT exchange with loss of the user information in D-U-ABORT service. Collisions of DAB APDUs are governed by the A-U-ABORT service.

6.5 *Capability*

6.5.1 *Purpose*

It supports the D-CAPABILITY service.

6.5.2 *APDUs used*

The DTAM capability procedure uses the D-CAPABILITY-REQ (DCPQ) and the D-CAPABILITY RESP (DCPR) APDUS

6.5.2.1 *DCPQ APDU*

The fields of the DCPQ APDU are listed in Table 8/T.433.

TABLE 8/T.433
DCPQ APDU fields

Field name	Presence	Source	Sink
Application capabilities			
Document application profile	U	request	indication
Document architecture class	U	request	indication
Non basic structural characteristics	U	request	indication
Non basic document characteristics	U	request	indication
Operational application profile	U	request	indication
Storage capacity	U	request	indication
User information	U	request	indication

6.5.2.2 DCPR APDU

The fields of the DCPR APDU are listed in Table 9/T.433.

TABLE 9/T.433
DCPR APDU fields

Field name	Presence	Source	Sink
Application capabilities			
Document application profile	U	response	confirmation
Document architecture class	U	response	confirmation
Non basic structural characteristics	U	response	confirmation
Non basic document characteristics	U	response	confirmation
Operational application profile	U	response	confirmation
Storage capacity	U	response	confirmation
Capability result	U	response	confirmation
User information	U	response	confirmation

6.5.3 DTAM capability procedure

6.5.3.1 DTAM Capability Procedure mapped onto Presentation Service (Normal Mode)

This procedure is driven by the following events:

- a D-CAPABILITY request primitive from the requestor;
- a DCPQ APDU as User Data on a P-CAPAB-DATA indication primitive;

- c) a D-CAPABILITY response primitive from the responder; and
- d) a P-CAPAB-DATA confirm primitive (that may contain a DCPR APDU).

6.5.3.1.1 *D-CAPABILITY request primitive*

6.5.3.1.1.1 The requesting DTAM-PM forms a DCPQ APDU from parameter values of the D-CAPABILITY request primitive. It issues a P-CAPAB-DATA request primitive. The User Data parameter of the P-CAPAB-DATA request primitive contains the DCPQ APDU.

6.5.3.1.1.2 The requesting DTAM-PM waits for a primitive from the Presentation service-provider, and does not accept any other primitive from the requestor other than a D-U-ABORT request primitive.

6.5.3.1.2 *DCPQ APDU*

6.5.3.1.2.1 The responding DTAM-PM receives a DCPQ APDU from its peer as User Data on a P-CAPAB-DATA indication primitive.

6.5.3.1.2.2 In order that the DCPQ APDU may always be acceptable to the responding DTAM-PM, it issues a D-CAPABILITY indication primitive to the responder. The D-CAPABILITY indication primitive parameters are derived from the DCPQ APDU. The DTAM-PM waits for a D-CAPABILITY response primitive from the responder and does not accept any other primitives from the responder except for the D-U-ABORT request primitive.

6.5.3.1.3 *D-CAPABILITY response primitive*

6.5.3.1.3.1 When the DTAM-PM receives the D-CAPABILITY response primitive, the Result parameter specifies whether the responder has accepted or rejected the DTAM capability requested. The DTAM-PM forms a DCPR APDU using the D-CAPABILITY response primitive parameters. The DCPR APDU is sent as the User Data parameter on the P-CAPAB-DATA response primitive.

6.5.3.1.3.2 If the responder accepted the DTAM capability request, the Capability Result field of the outgoing DCPR APDU also specifies the appropriate acceptance value. The DTAM capability is negotiated.

6.5.3.1.3.3 If the responder rejected the DTAM capability request, the Result field of the outgoing DCPR APDU contains the appropriate rejection value. The DTAM capability is not established.

6.5.3.1.4 *P-CAPAB-DATA confirm primitive*

6.5.3.1.4.1 The requesting DTAM-PM receives a P-CAPAB-DATA confirm primitive. The following situations are possible:

- a) the DTAM capability has been accepted; or
- b) the responder has rejected the DTAM capability requested by the requestor.

6.5.3.1.4.2 If the DTAM capability was accepted, the Capability Result field of the DCPR APDU specifies the appropriate acceptance value. The requesting DTAM-PM issues a D-CAPABILITY confirm primitive to the requestor based on parameters from the DCPR APDU. The D-CAPABILITY confirm primitive Capability Result Parameter specifies the appropriate acceptance value. The DTAM capability is negotiated.

6.5.3.1.4.3 If the DTAM capability was rejected by the responder, the Capability Result field of the DCPR APDU on the P-CAPAB-DATA confirm primitive indicates the reason for rejection. The requesting DTAM-PM issues a D-CAPABILITY confirm primitive to the requestor based on parameters from the DCPR APDU. The D-CAPABILITY confirm primitive Capability Result parameter contains the appropriate rejection value. The DTAM capability is not established.

6.5.3.2 *DTAM capability procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

- a) a D-CAPABILITY request primitive from the requestor;
- b) a DCPQ APDU as User Data on an S-CAPAB-DATA indication primitive;
- c) a D-CAPABILITY response primitive from the responder; and
- d) an S-CAPAB-DATA confirm primitive (that may contain a DCPR APDU).

6.5.3.2.1 *D-CAPABILITY request primitive*

6.5.3.2.1.1 The requesting DTAM-PM forms a DCPQ APDU from parameter values of the D-CAPABILITY request primitive, and issues an S-CAPAB DATA request primitive. The User Data parameter of the S-CAPAB DATA request primitive contains the DCPQ APDU.

6.5.3.2.1.2 The requesting DTAM-PM waits for a primitive from the Session service-provider, and does not accept any other primitive from the requestor other than a D-U-ABORT request primitive.

6.5.3.2.2 *DCPQ APDU*

6.5.3.2.2.1 The responding DTAM-PM receives a DCPQ APDU from its peer as User Data on an S-CAPAB DATA indication primitive.

6.5.3.2.2.2 In order that the S-CAPAB DATA indication primitive and its DCPQ APDU may always be acceptable to the responding DTAM-PM, it issues a D-CAPABILITY indication primitive to the responder. The D-CAPABILITY indication primitive parameters are derived from the DCPQ APDU. The DTAM-PM waits for a D-CAPABILITY response primitive from the responder and does not accept any other primitives from the responder except for the D-ABORT request primitive.

6.5.3.2.3 *D-CAPABILITY response primitive*

6.5.3.2.3.1 When the DTAM-PM receives the D-CAPABILITY response primitive, the parameters specified in its response primitive contain the Application Capabilities available at the responder. There is no way to issue the result of the capability negotiation explicitly. The DTAM-PM forms a DCPR APDU using the D-CAPABILITY response primitive parameters, and the DCPR APDU is sent as the User Data parameter on the S-CAPAB DATA response primitive.

6.5.3.2.3.2 In this way, the DTAM capability is negotiated by exchanging the Application Capabilities parameters available at the responder.

6.5.3.2.4 *S-CAPAB DATA confirm primitive*

6.5.3.2.4.1 The requesting DTAM-PM receives an S-CAPAB DATA confirm primitive. The DTAM capability is always negotiated by exchanging the Application Capabilities parameters.

6.5.3.2.4.2 If the DTAM capability was accepted, the requesting DTAM-PM issues a D-CAPABILITY confirm primitive to the requestor based on parameters from the DCPR APDU. The final decision of DTAM capability used in the transmission of a document will be made by the requesting DTAM-PM.

6.5.4 *Use of the DCPQ/DCPR APDU fields*

The DCPQ APDU and DCPR APDU fields are used as follows.

6.5.4.1 *Application capabilities*

This is the Application Capabilities parameter value from the D-CAPABILITY request/response primitives. It appears as the Application Capabilities parameter value of the D-CAPABILITY indication/confirm primitives respectively. This parameter consists of the following sub-parameters.

6.5.4.1.1 *Document application profile*

The value of this parameter is either an Octet String or ASN.1 object identifiers. The Octet String designates the document application profile in line with the Recommendation T.73 (Document Application Profile – T.73). the ASN.1 object identifier must conform to the rules specified in ISO 8824 and designate an application profile defined in accordance with the rules specified in Recommendation T.411 (Document Application Profiles).

6.5.4.1.2 *Document architecture class*

The value of this parameter is "formatted".

6.5.4.1.3 *Non basic document characteristics*

The value of this parameter is any combination of Non Basic Document Characteristics defined in Recommendation T.414.

6.5.4.1.4 *Non basic structural characteristics*

The value of this parameter is any combination of Non Basic Structural Characteristics defined in Recommendation T.414.

6.5.4.1.5 *Operational application profile*

The detailed specification of Operational Application Profile is for further study.

6.5.4.1.6 *Storage capacity*

See 6.2.4.8.

6.5.4.2 *Capability result*

If the DCPQ APDU was rejected by the responder, this field is supplied by the responder and is the Capability Result parameter from the D-CAPABILITY response primitive. In this situation, it appears as the Capability Result parameter on the D-CAPABILITY confirm primitive. This field can take one of the following:

- confirmation that all the requested capabilities are available at the DTAM responder;
- a list of the requested capabilities that are available at the DTAM responder;
- a complete list of non-basic receiving capabilities;
- an indication that no extended capabilities are available in the DTAM responder, or that none of the capabilities requested by the requestor are available.

6.5.4.3 *User information*

This is the User Information parameter from the D-CAPABILITY request and response primitive. It appears as the User Information parameter of the D-CAPABILITY indication and confirm primitive, if issued.

6.6 *Document bulk transfer*

6.6.1 *Purpose*

6.6.1.1 The document bulk transfer is used to convey the document which contains ODA and Operational Structure to the remote DTAM user. The requestor who requests the remote Document bulk transfer should have a data taken in an appropriate manner. It supports the D-TRANSFER services.

6.6.1.2 In this situation, either the Reliable Transfer Mode 1 or Mode 2 may be selected by the negotiation of functional units in the association establishment phase.

6.6.1.3 If the reliable transfer functional unit is not selected, the RTSE service will be used. The use of RTSE is for further study.

6.6.1.4 The Document Bulk Transfer is composed of two different sets of procedures depending on the Reliable Transfer Mode.

1) Reliable transfer mode 1

- a) Transfer Procedure for transmission of a complete document;
- b) Transfer-user-resume Procedure for retransmission of a partial document for resuming purposes. This procedure is controlled by the DTAM user;
- c) Transfer-interrupt Procedure to interrupt the transmission of a document in case of error;
- d) Transfer-discard Procedure to interrupt the transmission of a document in case of error and indicate that the part of the document already transmitted has to be deleted.

In Reliable Transfer Mode 1, the Transfer-interrupt and the Transfer-discard Procedures result in a D-TRANSFER indication/confirmation to the DTAM user to indicate the failure of the transfer. The user is then responsible for initiating a new transfer (complete or partial document).

Figures A-1/T.433 and A-2/T.433 illustrate the basic protocol sequences for the Reliable Transfer Mode 1.

2) Reliable transfer mode 2

- a) Transfer Procedure (see above, Reliable Transfer, Mode 1);
- b) Transfer-resume Procedure for retransmission of a partial document. This procedure is completely controlled by the DTAM-PM;
- c) Transfer-interrupt Procedure (see above Reliable Transfer, Mode 1);
- d) Transfer-discard Procedure (see above Reliable Transfer, Mode 1);
- e) Association-recovery Procedure (for further study).

In Reliable Transfer Mode 2, following the Transfer-interrupt and Transfer-discard Procedures, the DTAM-PM initiates a new Transfer Procedure or a Transfer-resume Procedure. Attempts to transfer or retransfer the document may not be performed by the DTAM-PM after the transfer time is out. The transfer time-out may result to discard the document and to abort the procedure.

Figures A-3/T.433 and A-4/T.433 illustrate the basic protocol sequences for the Reliable Transfer Mode 2.

In the Transparent Mode under Session Service environment, only Reliable Transfer Mode 1 is used.

In the Normal Mode under the OSI environment, both Reliable Transfer Modes 1 and 2 are available.

6.6.2 APDUs used

6.6.2.1 No APDUs are used in this procedure. The Document Information corresponds to a D-TRANSFER request service primitive. There is no D-TRANSFER REQ APDU as such.

6.6.2.2 Each Document Information, conveyed in a D-TRANSFER request, constitutes an Activity. For each application association, at most one Activity or one interrupted Activity awaiting resumption may exist at any one time.

6.6.2.3 The Document Information, which consists of one or more interchange-data-elements as defined in § 9.6.1.1 of Recommendation T.432, is segmented and reassembled into/from one or more segments. Each segment consists of one or more groups of interchange-data-elements and is transferred by the Presentation/Session data transfer services.

6.6.2.4 A Document Information is transferred as a single User Data of the Presentation/Session data transfer services if checkpointing is not used within the Document Information, otherwise, the Document Information is transferred as a series of Presentation/Session data transfer services primitives. The concatenation of the user-data values of Presentation/Session data transfer services is Document Information. An example of document segmenting mechanism is given in Figure 2/T.433.

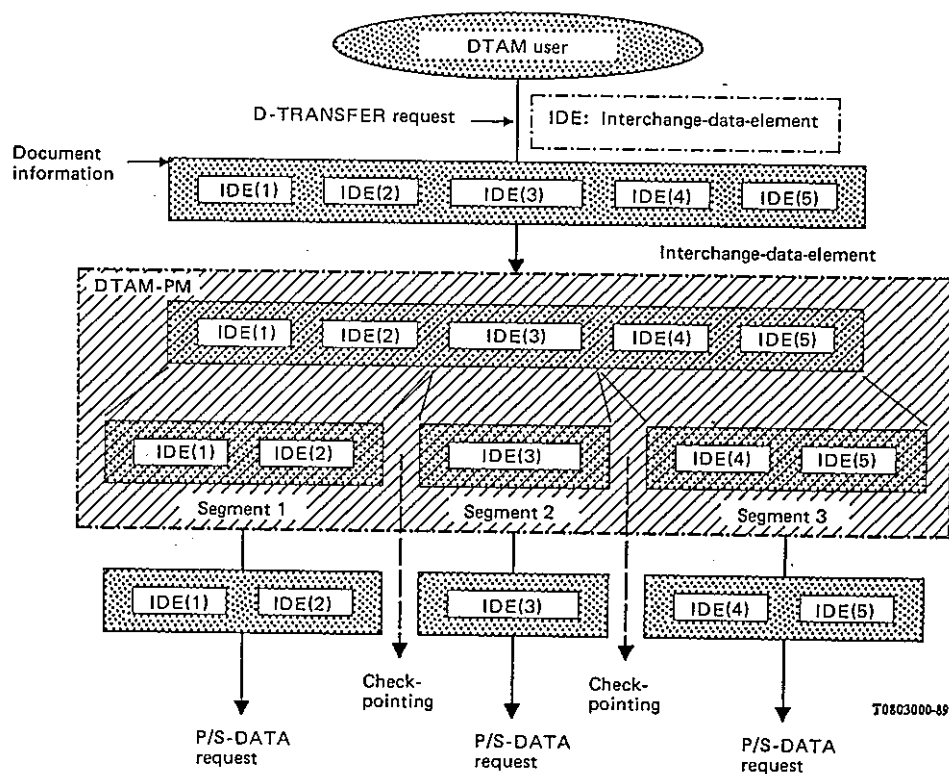


FIGURE 2/T.433

An example of document segmenting mechanism

6.6.3 Transfer procedure

This procedure is used to transfer a complete document.

6.6.3.1 Transfer procedure mapped onto presentation service (normal mode)

This procedure is driven by the following events:

- a D-TRANSFER request primitive from the requestor (sender of document);
- a P-ACTIVITY-START indication primitive, followed by one or more interchange-data-elements as user-data of P-DATA indication primitives each, except the last, followed by a P-MINOR-SYNCHRONIZE indication primitive;

- c) a P-MINOR-SYNCHRONIZE confirm primitive;
- d) a P-ACTIVITY-END indication primitive;
- e) a P-ACTIVITY-END confirm primitive;
- f) a Transfer Time-out.

Note – In the case of multiple documents transmission within one association, the above procedure will be applied repeatedly.

6.6.3.1.1 *D-TRANSFER request primitive*

6.6.3.1.1.1 If the requesting DTAM-PM possesses the Data Token and receives a D-TRANSFER request from the requestor, Document Information in the D-TRANSFER request primitive, which has an abstract form, is segmented by the group (segment) of interchange-data-elements. The segmenting unit (e.g. page, block) depends upon the characteristics of the DTAM-PM. The segmented abstract form is then transformed into the User Data in P-DATA.

6.6.3.1.1.2 The parameter "Document Information Type" contained in the D-TRANSFER request should indicate "transfer of a document from its beginning", and the requesting DTAM-PM issues a P-ACTIVITY-START request primitive and may start transmitting the first segment of interchange-data- elements in a P-DATA request primitive immediately after the P-ACTIVITY-START request primitive is issued, since the P-ACTIVITY-START service is not a confirmed service.

6.6.3.1.1.3 If the segment of interchange-data-elements transferred is not the last in a series of those segments, the requesting DTAM-PM inserts a checkpoint by issuing a P-MINOR-SYNCHRONIZE request primitive. The requesting DTAM-PM uses only the "explicit confirmation expected" type of minor synchronization. The requesting DTAM-PM may issue further P-DATA request primitives and P-MINOR- SYNCHRONIZE request primitives unless the agreed Window-size has been reached.

6.6.3.1.1.4 P-Minor-Synchronization Points shall be located at the end of each segment of interchange-data-elements. Additional Minor Synchronization Point can be requested depending on the evaluation of the storage capacity of the sink and the amount of data to be transmitted. This additional Minor Synchronization Point shall only be located at the end of any interchange-data- elements and not within the element.

6.6.3.1.1.5 If the segment of interchange-data-elements is the only one, or the last in a series of segments of interchange-data-elements, the requesting DTAM-PM issues a P-ACTIVITY-END request primitive. All data transfer must take place within an activity.

6.6.3.1.2 *P-ACTIVITY-START indication primitive, P-DATA PDUs, and P-MINOR-SYNCHRONIZE indication primitives*

6.6.3.1.2.1 The responding DTAM-PM receives a P-ACTIVITY-START indication primitive, indicating the start of transfer of Document Information. The responding DTAM-PM receives a P-MINOR-SYNCHRONIZE indication primitive. If the responding DTAM-PM has secured the segment of interchange-data-elements, it issues a P-MINOR-SYNCHRONIZE response primitive.

6.6.3.1.3 *P-MINOR-SYNCHRONIZE confirm primitive*

6.6.3.1.3.1 When the requesting DTAM-PM receives a P-MINOR-SYNCHRONIZE confirm primitive, it assumes that the responding DTAM-PM has secured the segments of interchange-data-elements up to that point.

6.6.3.1.3.2 The requesting DTAM-PM may issue further P-DATA request primitives and P-MINOR-SYNCHRONIZE request primitives unless the agreed Window-size has been reached. The window is advanced when a P-MINOR-SYNCHRONIZE confirm primitive is received by the requesting DTAM-PM.

6.6.3.1.3.3 When a complete Document Information has been transmitted, the requesting DTAM-PM issues a P-ACTIVITY-END request primitive.

6.6.3.1.4 *P-ACTIVITY-END indication primitive*

6.6.3.1.4.1 A P-ACTIVITY-END Indication primitive indicates to the responding DTAM-PM that a complete Document Information has been transferred.

6.3.1.4.2 If the responding DTAM-PM has secured the complete Document Information, it issues a D-TRANSFER indication primitive to the responder, and issues a P-ACTIVITY-END response primitive.

6.6.3.1.4.3 The responding DTAM-PM records the Session-connection-identifier and the Activity Identifier of the last Document Information which it completely secured for association-recovery purposes.

6.6.3.1.5 *P-ACTIVITY-END confirm primitive*

6.6.3.1.5.1 An activity end is an implicit major synchronization point and once successfully confirmed by means of a P-ACTIVITY-END confirm primitive, it indicates to the requesting DTAM-PM that the Document Information has been secured by the responding DTAM-PM. The requesting DTAM-PM may then delete the transferred Document Information.

6.6.3.1.5.2 When the requesting DTAM-PM receives the P-ACTIVITY-END confirm primitive, it issues a D-TRANSFER confirm primitive with a Result parameter value of "document-information-transferred" to the requestor.

6.6.3.1.6 *Transfer time-out (only for reliable transfer mode 2)*

6.6.3.1.6.1 If a Document Information has not been transferred within the time specified in the Transfer-time parameter of the D-TRANSFER request primitive (that is, the requesting DTAM-PM has not received the P-ACTIVITY-END confirm primitive), the requesting DTAM-PM performs the transfer-discard procedure (see § 6.6.6) followed by the transfer-abort procedure (see § 6.4.3.1.4).

6.6.3.1.6.2 If during the transfer-discard procedure the requesting DTAM-PM does not receive a P-ACTIVITY-DISCARD confirm primitive within a (locally specified) reasonable time, the requesting DTAM-PM performs the transfer-abort procedure followed by the DTAM provider-abort procedure.

6.6.3.2 Transfer procedure mapped onto session service (transparent mode)

This procedure is driven by the following events:

- a) a D-TRANSFER request primitive from the requestor (sender of document);
- b) an S-ACTIVITY-START indication primitive, followed by one or more interchange-data-elements as user-data of S-DATA indication primitives each, except the last, followed by an S-MINOR-SYNCHRONIZE indication primitive;
- c) an S-MINOR-SYNCHRONIZE confirm primitive;
- d) an S-ACTIVITY-END indication primitive;
- e) an S-ACTIVITY-END confirm primitive;

Note – In the case of multiple document transmission within one association, the above procedure will be applied repeatedly.

6.6.3.2.1 *D-TRANSFER request primitive*

6.6.3.2.1.1 If the requesting DTAM-PM possesses the Data Token and receives a D-TRANSFER request from the requestor, Document Information in the D-TRANSFER request primitive which has an abstract form is segmented by the group (segment) of interchange-data-elements. The segmenting unit (e.g. page, block) depends upon the characteristics of the DTAM-PM. The segmented abstract form is then transformed into the User Data in S-DATA.

6.6.3.2.1.2 The parameter "Document Information Type" contained in the D-TRANSFER request should indicate the "transfer of a document from its beginning", and the requesting DTAM-PM issues an S-ACTIVITY-START request primitive and may start transmitting the first segment of interchange-data-elements in an S-DATA request primitive immediately after the S-ACTIVITY-START request primitive is issued, since the S-ACTIVITY-START service is not a confirmed service. All data transfer should take place within an activity.

6.6.3.2.1.3 If the segment of interchange-data-elements transferred is not the last in a series of those segments, the requesting DTAM-PM inserts a checkpoint by issuing an S-MINOR-SYNCHRONIZE request primitive. The requesting DTAM-PM uses only the "explicit confirmation expected" type of minor synchronization. The requesting DTAM-PM may issue further S-DATA request primitives and S-MINOR-SYNCHRONIZE request primitives unless the agreed Window-size has been reached.

6.6.3.2.1.4 S-Minor-Synchronization Points shall be located at the end of each segment of interchange-data-elements. Additional Minor Synchronization Points can be requested depending on the evaluation of the storage capacity of the sink and the amount of data to be transmitted. This additional Minor Synchronization Points shall only be located at the end of any interchange-data-elements and not within the element.

6.6.3.2.1.5 If the segment of interchange-data-elements is the only one, or the last in a series of segments of interchange-data-elements, the requesting DTAM-PM issues an S-ACTIVITY-END request primitive. All data transfer must take place within an activity.

6.6.3.2.2 *S-ACTIVITY-START indication primitive, S-DATA PDUs, and S-MINOR-SYNCHRONIZE indication primitives*

6.6.3.2.2.1 The responding DTAM-PM receives an S-ACTIVITY-START indication primitive, indicating the start of transfer of Document Information. The responding DTAM-PM receives an S-MINOR-SYNCHRONIZE indication primitive. If the responding DTAM-PM has secured the segment of interchange-data-elements, it issues an S-MINOR-SYNCHRONIZE response primitive.

6.6.3.2.3 *S-MINOR-SYNCHRONIZE confirm primitive*

6.6.3.2.3.1 When the requesting DTAM-PM receives an S-MINOR-SYNCHRONIZE confirm primitive, it assumes that the responding DTAM-PM has secured the segments of interchange-data-elements up to that point.

6.6.3.2.3.2 The requesting DTAM-PM may issue further S-DATA request primitives and S-MINOR-SYNCHRONIZE request primitives unless the agreed Window-size has been reached. The window is advanced when an S-MINOR-SYNCHRONIZE confirm primitive is received by the requesting DTAM-PM.

6.6.3.2.3.3 When a complete Document Information has been transmitted, the requesting DTAM-PM issues an S-ACTIVITY-END request primitive.

6.6.3.2.4 *S-ACTIVITY-END indication primitive*

6.6.3.2.4.1 An S-ACTIVITY-END Indication primitive indicates to the responding DTAM-PM that a complete Document Information has been transferred.

6.6.3.2.4.2 If the responding DTAM-PM has secured the complete Document Information, it issues a D-TRANSFER indication primitive to the responder, and issues an S-ACTIVITY-END response primitive.

6.6.3.2.5 *S-ACTIVITY-END confirm primitive*

6.6.3.2.5.1 An activity end is an implicit major synchronization point and once successfully confirmed by means of an S-ACTIVITY-END confirm primitive, it indicates to the requesting DTAM-PM that the Document Information has been secured by the responding DTAM-PM. The requesting DTAM-PM may then delete the transferred Document Information.

6.6.3.2.5.2 When the requesting DTAM-PM receives the S-ACTIVITY-END confirm primitive, it issues a D-TRANSFER confirm primitive with a Result parameter value of "document-information-transferred" to the requestor.

6.6.4 *Transfer-user-resume procedure*

This procedure is used to resume transferring the part of the document which has not been transferred at the previous transmission.

6.6.4.1 *Transfer-user-resume procedure mapped onto presentation service (normal mode)*

This procedure is driven by the following events:

- a) a D-TRANSFER request primitive from the requestor (sender of document);
- b) a P-ACTIVITY-RESUME indication primitive, followed by one or more interchange-data-elements as user-data of P-DATA indication primitives each, except the last, followed by a P-MINOR-SYNCHRONIZE indication primitive;
- c) a P-MINOR-SYNCHRONIZE confirm primitive;
- d) a P-ACTIVITY-END indication primitive;
- e) a P-ACTIVITY-END confirm primitive.

6.6.4.1.1 *D-TRANSFER request primitive*

6.6.4.1.1.1 If the requesting DTAM-PM possesses the Data Token and receives a D-TRANSFER request from the requestor, Document Information in the D-TRANSFER request primitive, which has an abstract form, is segmented by the group (segment) of interchange-data-elements. The segmenting unit (e.g. page, block) depends upon the characteristics of the DTAM-PM. The segmented abstract form is then transformed into the User Data in P-DATA.

6.6.4.1.1.2 The parameter "Document Information Type" contained in the D-TRANSFER request should indicate "transfer of a document from a synchronization point", and the requesting DTAM-PM issues a P-ACTIVITY-RESUME request primitive and may continue the transfer procedure by issuing a P-DATA request primitive for the segment of interchange-data-elements following the last confirmed checkpoint. The checkpoint information is from the parameter "Synchronization Point" in the D-TRANSFER request primitive.

6.6.4.1.1.3 Another detailed procedure is followed by the § 6.6.3.1.1.3, 6.6.3.1.1.4 and 6.6.3.1.1.5.

6.6.4.1.2 *P-ACTIVITY-RESUME indication primitive, P-DATA PDUs, and P-MINOR-SYNCHRONIZE indication primitives*

6.6.4.1.2.1 The responding DTAM-PM receives a P-ACTIVITY-RESUME indication primitive, indicating the start of transfer of Document Information. The responding DTAM-PM receives a P-MINOR-SYNCHRONIZE indication primitive. If the responding DTAM-PM has secured the segment of interchange-data-elements, it issues a P-MINOR-SYNCHRONIZE response primitive.

6.6.4.1.3 *P-MINOR-SYNCHRONIZE confirm primitive*

6.6.4.1.3.1 The detailed procedure is followed by the § 6.6.3.1.3.1, 6.6.3.1.3.2 and 6.6.3.1.3.3.

6.6.4.1.4 *P-ACTIVITY-END indication primitive*

6.6.4.1.4.1 The detailed procedure is followed by the § 6.6.3.1.4.1, 6.6.3.1.4.2 and 6.6.3.1.4.3.

6.6.4.1.5 *P-ACTIVITY-END confirm primitive*

6.6.4.1.5.1 The detailed procedure is followed by the § 6.6.3.1.5.1 and 6.6.3.1.5.2.

6.6.4.2 Transfer-user-resume procedure mapped session service (transparent mode)

This procedure is driven by the following events:

- a) a D-TRANSFER request primitive from the requestor (sender of document);
- b) an S-ACTIVITY-RESUME indication primitive, followed by one or more interchange-data-elements as user-data of S-DATA indication primitives each, except the last, followed by an S-MINOR-SYNCHRONIZE indication primitive;
- c) an S-MINOR-SYNCHRONIZE confirm primitive;
- d) an S-ACTIVITY-END indication primitive;
- e) an S-ACTIVITY-END confirm primitive.

6.6.4.2.1 *D-TRANSFER request primitive*

6.6.4.2.1.1 If the requesting DTAM-PM possesses the Data Token and receives a D-TRANSFER request from the requestor, Document Information in the D-TRANSFER request primitive, which has an abstract form, is segmented by the group (segment) of interchange-data-elements. The segmenting unit (e.g. page, block) depends upon the characteristics of the DTAM-PM. The segmented abstract form is then transformed into the User Data in P-DATA.

6.6.4.2.1.2 The parameter "Document Information Type" contained in the D-TRANSFER request should indicate the "transfer of a document from a synchronization point", and the requesting DTAM-PM issues an S-ACTIVITY-RESUME request primitive and may continue the transfer procedure by issuing a S-DATA request primitive for the segment of interchange-data-elements following the last confirmed checkpoint. The checkpoint information is from the parameter "Synchronization Point" in the D-TRANSFER request primitive.

6.6.4.2.1.3 Another detailed procedure is followed by the § 6.6.3.2.1.3, 6.6.3.2.1.4 and 6.6.3.2.1.5.

6.6.4.2.2 *S-ACTIVITY-RESUME indication primitive, S-DATA PDUs, and S-MINOR-SYNCHRONIZE indication primitives*

6.6.4.2.2.1 The responding DTAM-PM receives an S-ACTIVITY-RESUME indication primitive, indicating the start of transfer of Document Information. The responding DTAM-PM receives an S-MINOR-SYNCHRONIZE indication primitive. If the responding DTAM-PM has secured the segment of interchange- data-elements, it issues an S-MINOR-SYNCHRONIZE response primitive.

6.6.4.2.3 *S-MINOR-SYNCHRONIZE confirm primitive*

6.6.4.2.3.1 The detailed procedure is followed by the § 6.6.3.2.3.1, 6.6.3.2.3.2 and 6.6.3.2.3.3.

6.6.4.2.4 *S-ACTIVITY-END indication primitive*

6.6.4.2.4.1 The detailed procedure is followed by the § 6.6.3.2.4.1 and 6.6.3.2.4.2.

6.6.4.2.5 *S-ACTIVITY-END confirm primitive*

6.6.4.2.5.1 The detailed procedure is followed by the § 6.6.3.2.5.1 and 6.6.3.2.5.2.

6.6.5 *Transfer-interrupt*

6.6.5.1 Purpose

The transfer-interrupt procedure is used by the requesting DTAM-PM to handle a less severe (than those handled by the other error handling procedures) error situation during the transfer procedure, if at least one checkpoint was confirmed during the transfer procedure.

6.6.5.2 APDUs used

No APDUs are used in this procedure.

6.6.5.3 Transfer-interrupt procedure

6.6.5.3.1 *Transfer-interrupt procedure mapped onto presentation service (normal mode)*

This procedure is driven by the following events:

- a) a requesting DTAM-PM problem;
- b) a P-ACTIVITY-INTERRUPT indication primitive;
- c) a P-ACTIVITY-INTERRUPT confirm primitive.

6.6.5.3.1.1 Requesting DTAM-PM problem

6.6.5.3.1.1.1 If the requesting DTAM-PM detects a less severe problem and at least one checkpoint was confirmed during the transfer procedure, it issues a P-ACTIVITY-INTERRUPT request primitive with one of the following Reason parameter values:

- a) "non-specific error", if the problem was indicated by an exception reporting procedure,
- b) "local SS-User error", if the problem is a local requesting DTAM-PM problem.

6.6.5.3.1.2 P-ACTIVITY-INTERRUPT indication primitive

6.6.5.3.1.2.1 If the responding DTAM-PM receives a P-ACTIVITY-INTERRUPT indication primitive, it issues:

- a P-ACTIVITY-INTERRUPT response primitive followed by a D-TRANSFER indication in Reliable transfer mode 1;
- a P-ACTIVITY-INTERRUPT response primitive in Reliable transfer mode 2.

6.6.5.3.1.3 P-ACTIVITY-INTERRUPT confirm primitive

6.6.5.3.1.3.1 If the requesting DTAM-PM receives a P-ACTIVITY-INTERRUPT confirm primitive, it issues:

- D-TRANSFER confirmation to the requesting DTAM user in Reliable transfer mode 1;
- the transfer-resumption procedure in Reliable transfer mode 2.

6.6.5.3.2 *Transfer-interrupt procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

- a) a requesting DTAM-PM problem;
- b) a S-ACTIVITY-INTERRUPT indication primitive;
- c) a S-ACTIVITY-INTERRUPT confirm primitive.

6.6.5.3.2.1 Requesting DTAM-PM problem

6.6.5.3.2.1.1 If the requesting DTAM-PM detects a less severe problem and at least one checkpoint was confirmed during the transfer procedure, it issues a S-ACTIVITY-INTERRUPT request primitive with one of the following Reason parameter values:

- a) "non-specific error", if the problem was indicated by an exception reporting procedure;
- b) "local SS-User error", if the problem is a local requesting DTAM-PM problem.

6.6.5.3.2.2 S-ACTIVITY-INTERRUPT indication primitive

6.6.5.3.2.2.1 If the responding DTAM-PM receives an S-ACTIVITY-INTERRUPT confirm primitive, it issues an S-ACTIVITY-INTERRUPT response primitive followed by D-TRANSFER indication.

6.6.5.3.2.3 S-ACTIVITY-INTERRUPT confirm primitive

6.6.5.3.2.3.1 If the requesting DTAM-PM receives an S-ACTIVITY-INTERRUPT confirm primitive, it issues D-TRANSFER confirmation to the requesting DTAM user.

6.6.6 *Transfer-discard*

6.6.6.1 Purpose

The transfer-discard procedure is used by the requesting DTAM-PM to escape from a more severe (than those handled by the transfer-interrupt procedure) error situation, or a less severe error situation if no checkpoint was confirmed, during the transfer procedure.

6.6.6.2 APDUs used

No APDUs are used in this procedure.

6.6.6.3 Transfer-discard procedure

6.6.6.3.1 *Transfer-discard procedure mapped onto presentation service (normal mode)*

This procedure is driven by the following events:

- a) a requesting DTAM-PM problem;
- b) a P-ACTIVITY-DISCARD indication primitive;
- c) a P-ACTIVITY-DISCARD confirm primitive.

6.6.6.3.1.1 Requesting DTAM-PM problem

6.6.6.3.1.1.1 If the requesting DTAM-PM detects a more severe problem, or a less severe problem if no checkpoint was confirmed during the transfer procedure, it issues a P-ACTIVITY-DISCARD request primitive with one of the following reason parameter values:

- a) "non-specific error", if the problem was indicated by an error reporting procedure;
- b) "local SS-User error", or "unrecoverable procedural error", if the problem is a local requesting DTAM-PM problem.

6.6.6.3.1.2 P-ACTIVITYDISCARD indication primitive

6.6.6.3.1.2.1 If the responding DTAM-PM receives a P-ACTIVITY-DISCARD indication primitive, it issues:

- a P-ACTIVITY-DISCARD response primitive followed by D-TRANSFER indication in reliable transfer mode 1;
- a P-ACTIVITY-DISCARD response primitive in reliable transfer mode 2.

6.6.6.3.1.2.2 The responding DTAM-PM deletes all knowledge and contents of the associated DTAM user information (segments of document information) so far received.

6.6.6.3.1.2.3 If the responding DTAM-PM has already issued a D-TRANSFER indication primitive, it performs the association-abort procedure. The abort-reason field of the DAB APDU is "transfer-completed".

6.6.6.3.1.3 P-ACTIVITY-DISCARD confirm primitive

6.6.6.3.1.3.1 If the requesting DTAM-PM receives a P-ACTIVITYDISCARD confirm primitive, it issues:

- D-TRANSFER confirmation to the requesting DTAM user in reliable transfer mode 1;
- the transfer-retry procedure in reliable transfer mode 2.

6.6.6.3.2 *Transfer-discard procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

- a) a requesting DTAM-PM problem;
- b) an S-ACTIVITY-DISCARD indication primitive;
- c) an S-ACTIVITY-DISCARD confirm primitive.

6.6.6.3.2.1 Requesting DTAM-PM problem

6.6.6.3.2.1.1 If the requesting DTAM-PM detects a more severe problem, or a less severe problem if no checkpoint was confirmed during the transfer procedure, it issues an S-ACTIVITY-DISCARD request primitive with one of following reason parameter values:

- a) "non-specific error", if the problem was indicated by an error reporting procedure;
- b) "local SS-User error", or "unrecoverable procedural error", if the problem is a local requesting DTAM-PM problem.

6.6.6.3.2.2 S-ACTIVITY-DISCARD indication primitive

- 6.6.6.3.2.2.1 If the responding DTAM-PM receives an S-ACTIVITY-DISCARD indication primitive, it issues an S-ACTIVITY-DISCARD response primitive followed by D-TRANSFER indication in reliable transfer mode 1.
- 6.6.6.3.2.2.2 The responding DTAM-PM deletes all knowledge and contents of the associated DTAM user information (segments of document information) so far received.
- 6.6.6.3.2.2.3 If the responding DTAM-PM has already issued a D-TRANSFER indication primitive, it performs the session-abort procedure by issuing an S-U-ABORT request.
- 6.6.6.3.2.3 S-ACTIVITY-DISCARD confirm primitive
- If the requesting DTAM-PM receives an S-ACTIVITY-DISCARD confirm primitive, it issues D-TRANSFER confirmation to the requesting DTAM user in reliable transfer mode 1.
- 6.6.7 *Transfer-resumption*
- 6.6.7.1 *Purpose*
- The transfer-resumption procedure is used in the reliable transfer mode 2 by the requesting DTAM-PM to recover from:
- an error situation handled by the transfer-interrupt procedure, or
 - an error situation handled by the association-abort procedure during a transfer procedure. In this case the transfer-resumption procedure is performed after an association-recovery procedure is successfully performed. If no checkpoint was confirmed in the interrupted transfer procedure, the transfer-discard procedure followed by the transfer-retry procedure are performed.
- 6.6.7.2 *APDUs used*
- No APDUs are used in this procedure.
- 6.6.7.3 *Transfer-resumption procedure*
- 6.6.7.3.1 *Transfer-resumption procedure mapped onto presentation service (normal mode)*
- This procedure is driven by the following events:
- the resumption of an interrupted activity;
 - a P-ACTIVITY-RESUME indication primitive.
- After these events, the transfer procedure is used to continue (see § 6.6.3).
- 6.6.7.3.1.1 *Resumption of an interrupted activity*
- 6.6.7.3.1.1.1 The requesting DTAM-PM issues a P-ACTIVITY-RESUME request primitive with parameters that link the resumed activity to the previously interrupted one.
- 6.6.7.3.1.1.2 After the requesting DTAM-PM has issued the P-ACTIVITY-RESUME request primitive and at least one checkpoint was confirmed in the interrupted transfer procedure, it continues the transfer procedure by issuing a P-DATA request primitive for the segment of interchange-data-elements following the last confirmed checkpoint. If no checkpoint was confirmed in the interrupted transfer procedure, the transfer-discard procedure followed by the transfer-retry procedure are performed.
- 6.6.7.3.1.2 *P-ACTIVITY-RESUME indication primitive*
- 6.6.7.3.1.2.1 If the responding DTAM-PM receives a P-ACTIVITY-RESUME indication primitive, it checks the old activity identifier and the old session connection identifier parameters of the P-ACTIVITY-RESUME indication primitive with the corresponding information (session-connection-identifier and activity identifier) recorded for the last completely secured transfer (see § 6.6.3.1.4.3).
- 6.6.7.3.1.2.2 If the information coincides with the recorded ones, the responding DTAM-PM either (a) responds correctly to the requesting DTAM-PM according to the normal transfer procedure, but discards the data it receives, and does not issue a D-TRANSFER indication primitive, or (b) performs the user-exception-report procedure with a Reason parameter value of "sequence error".
- 6.6.7.3.1.2.3 If the information does not coincide, the transfer-resumption procedure continues as for the transfer procedure with a P-DATA indication primitive for the segment of interchange-data-element following the last confirmed checkpoint.
- 6.6.7.3.1.2.4 If the responding DTAM-PM cannot resume the activity, the responding DTAM-PM performs the user-exception-report procedure.

6.6.8 Association-recovery

6.6.8.1 Purpose

The association-recovery procedure is used by the association-initiating DTAM-PM to recover from an error situation handled by the association-abort procedure or the association-provider-abort procedure.

This procedure is for further study.

6.7 Document unconfirmed manipulation

Document unconfirmed manipulation is used by the requestor to manipulate the constituents of ODA and operational structure which are commonly handled by both communication entities. Document unconfirmed manipulation consists of document create operation, document delete operation, document modify operation, document call operation and document rebuild operation.

6.7.1 Document create operation

6.7.1.1 Purpose

The document create operation procedure is used by the requestor of document manipulation to add the constituents of ODA and operational structure to a document without any confirmation of the create manipulation.

6.7.1.2 APDUs used

The document create operation procedure uses D-CREATE (DCR) APDU.

6.7.1.2.1 DCR APDU

The field of the DCR APDU is listed in Table 10/T.433.

TABLE 10/T.433

DCR APDU field

Field name	Presence	Source	Sink
Create information	M	req	ind

6.7.1.3 Document create operation procedure

6.7.1.3.1 Document create operation procedure mapped onto presentation service (normal mode)

Procedure is driven by the following events:

- a D-CREATE request primitive from the requestor;
- a DCR APDU as user data of a P-DATA indication primitive.

6.7.1.3.1.1 D-CREATE request primitive

6.7.1.3.1.1.1 If the requesting DTAM-PM receives a D-CREATE request primitive, a DCR APDU is formed from the parameter values of the D-CREATE request primitive and transferred as user-data of a P-DATA request primitive. This may be done outside of an activity.

6.7.1.3.1.2 DCR APDU

6.7.1.3.1.2.1 If the responding DTAM-PM receives the DCR APDU as user data of a P-DATA indication primitive, the responding DTAM-PM issues a D-CREATE indication primitive to the responder. The DCREATE indication primitive parameter is derived from the DCR APDU.

6.7.1.4 Use of the DCR APDU fields

The DCR APDU fields are used as specified below.

6.7.1.4.1 Create information

This is the create information parameter value from the D-CREATE request primitive. It appears as the create information parameter value of the D-CREATE indication primitive.

6.7.2 Document delete operation

6.7.2.1 Purpose

The document delete operation procedure is used by the requestor of document manipulation to delete the constituents of ODA and operational structure of an existing document without any confirmation of the delete operation.

6.7.2.2 APDUs used

The document delete operation procedure uses D-DELETE (DDL) APDU.

6.7.2.2.1 DDL APDU

The field of the DDL APDU is listed in Table 11/T.433.

TABLE 11/T.433

DDL APDU field

Field name	Presence	Source	Sink
Delete information	M	req	ind

6.7.2.3 Document delete operation procedure

6.7.2.3.1 Document delete operation procedure mapped onto presentation service (normal mode)

This procedure request is driven by the following events:

- a) a D-DELETE request primitive from the requestor;
- b) a DDL APDU as user data of P-DATA indication primitive.

6.7.2.3.1.1 D-DELETE request primitive

6.7.2.3.1.1.1 If the requesting DTAM-PM receives a D-DELETE request primitive, a DDL APDU is formed from the parameter values of D-DELETE request primitive and transferred as user-data of a P-DATA request primitive. This may be done outside of an activity.

6.7.2.3.1.2 DDL APDU

6.7.2.3.1.2.1 If the responding DTAM-PM receives the DDL APDU as user data of a P-DATA indication primitive, the responding DTAM-PM issues a D-DELETE indication primitive to the responder. The D-DELETE indication primitive parameter is derived from the DDL APDU.

6.7.2.4 Use of the DDL APDU fields

The DDL APDU fields are used as specified below.

6.7.2.4.1 Delete information

This is the delete information parameter value from the D-DELETE request primitive. It appears as the delete information parameter value of the D-DELETE indication primitive.

6.7.3 Document modify operation

6.7.3.1 Purpose

The document modify operation procedure is used by the requestor of document manipulation to modify the attributes of constituents of ODA and operational structure of an existing document without any confirmation of the modifying operation.

6.7.3.2 APDUs used

The document modify operation procedure uses D-MODIFY (DMD) APDU.

6.7.3.2.1 DMD APDU

The field of the DMD APDU is listed in Table 12/T.433.

TABLE 12/T.433

DMD APDU field

Field name	Presence	Source	Sink
Modify information	M	req	ind

6.7.3.3 Document modify operation procedure**6.7.3.3.1 Document modify operation procedure mapped onto presentation service (normal mode)**

This procedure is driven by the following events:

- a) a D-MODIFY request primitive from the requestor;
- b) a DMD APDU as user data of a P-DATA indication primitive.

6.7.3.3.1.1 D-MODIFY request primitive

6.7.3.3.1.1.1 If the requesting DTAM-PM receives a D-MODIFY request primitive, a DMD APDU is formed from the parameter values of D-MODIFY request primitive and transferred as user-data of a P-DATA request primitive. This may be done outside of an activity.

6.7.3.3.1.2 DMD APDU

6.7.3.3.1.2.1 If the responding DTAM-PM receives the DMD APDU as user data of a P-DATA indication primitive, the responding DTAM-PM issues a D-MODIFY indication primitive to the responder. The D-MODIFY indication primitive parameter is derived from the DMD APDU.

6.7.3.4 Use of the DMD APDU fields

The DMD APDU fields are used as specified below.

6.7.3.4.1 Modify information

This is the modify information parameter value from the D-MODIFY request primitive. It appears as the modify information parameter value of the D-MODIFY indication primitive.

6.7.4 Document call operation**6.7.4.1 Purpose**

The document call operation procedure is used by the requestor of document manipulation to address or to read an object of operational structure which contains a sequence of DTAM protocol data units (with some restrictions, i.e. that only D-CREATE, D-DELETE and D-MODIFY can appear in this sequence). These protocol data units are applicable to the existing document.

6.7.4.2 APDUs used

The document call operation procedure uses D-CALL (DCL) APDU.

6.7.4.2.1 DCL APDU

The field of the DCL APDU is listed in Table 13/T.433.

TABLE 13/T.433

DCL APDU field

Field name	Presence	Source	Sink
Call information	M	req	ind

6.7.4.3 Document call operation procedure**6.7.4.3.1 Document call operation procedure mapped onto presentation service (normal mode)**

This procedure is driven by the following events:

- a) a D-CALL request primitive from the requestor;

- b) a DCL APDU as user data of a P-DATA indication primitive.

6.7.4.3.1.1 *D-CALL request primitive*

6.7.4.3.1.1.1 If the requesting DTAM-PM receives a D-CALL request primitive, a DCL APDU is formed from the parameter values of D-CALL request primitive and transferred as user-data of a P-DATA request primitive. This may be done outside of an activity.

6.7.4.3.1.2 *DCL APDU*

6.7.4.3.1.2.1 If the responding DTAM-PM receives the DCL APDU as user data of a P-DATA indication primitive, the responding DTAM-PM issues a D-CALL indication primitive to the responder. The D-CALL indication primitive parameter is derived from the DCL APDU.

6.7.4.4 *Use of the DCL APDU fields*

The DCL APDU fields are used as specified below.

6.7.4.4.1 *Call information*

This is the call information parameter value from the D-CALL request primitive. It appears as the call information parameter value of the D-CALL indication primitive.

6.7.5 *Document rebuild operation*

(For further study.)

6.8 *Document confirmed manipulation*

(For further study.)

6.9 *Typed data transfer*

6.9.1 *Purpose*

Typed data transmission is used independent of the data token and is issued from both DTAM users when required.

6.9.2 *APDUs used*

The typed data transfer procedure uses D-TYPED-DATA (DTD) APDU.

6.9.2.1 *DTD APDU*

The field of the DTD APDU is listed in Table 14/T.433.

TABLE 14/T.433

DTD APDU field

Field name	Presence	Source	Sink
Typed data information	M	req	ind

6.9.3 *Typed data transfer procedure*

6.9.3.1 *Typed data transfer procedure mapped onto presentation service*

This procedure is driven by the following events:

- a D-TYPED-DATA request primitive from the requestor;
- a DTD APDU as user data of a P-TYPED-DATA indication primitive.

6.9.3.1.1 *D-TYPED-DATA request primitive*

6.9.3.1.1.1 If the requesting DTAM-PM receives a D-TYPED-DATA request primitive, a DTD APDU is formed from the parameter values of D-TYPED-DATA request primitive and transferred as user-data of a P-TYPED-DATA request primitive.

6.9.3.1.2 DTD APDU

6.9.3.1.2.1 If the responding DTAM-PM receives the DTD APDU as user data of a P-TYPED-DATA indication primitive, the responding DTAM-PM issues a D-TYPED-DATA indication primitive to the responder. The D-TYPED-DATA indication primitive parameter is derived from the DTD APDU.

6.9.4 Use of the DTD APDU fields

The DTD APDU fields are used as specified below.

6.9.4.1 Typed data information

This is the typed data information parameter value from the D-TYPED-DATA request primitive. It appears as the typed data information parameter value of the D-TYPED-DATA indication primitive.

6.10 Remote document access

(For further study.)

6.11 Remote document management

(For further study.)

6.12 Token control

6.12.1 Token please control

6.12.1.1 Purpose

The token please procedure is used by a requestor (receiver of documents) to request the token from the responder (sender of documents).

6.12.1.2 APDUs used

The token please procedure uses the D-TOKEN-PLEASE (DTP) APDU.

6.12.1.2.1 DTP APDU

The field of the DTP APDU is listed in Table 15/T.433.

TABLE 15/T.433

DTP APDU field

Field name	Presence	Source	Sink
Priority	U	req	ind

6.12.1.3 Token please procedure

6.12.1.3.1 Token please procedure mapped onto presentation service (normal mode)

This procedure is driven by the following events:

- a D-TOKEN-PLEASE request primitive from the requestor; and
- a DTP APDU as user data of a P-TOKEN-PLEASE indication primitive.

6.12.1.3.1.1 D-TOKEN-PLEASE request primitive

6.12.1.3.1.1.1 If the requesting DTAM-PM does not possess the token and receives a D-TOKEN-PLEASE request from the requestor, a DTP APDU is formed from the parameter value of the D-TOKEN-PLEASE request primitive and transferred as user data of a P-TOKEN-PLEASE request primitive. This may be done either inside or outside an activity.

6.12.1.3.1.2 DTP APDU

6.12.1.3.1.2.1 If the responding DTAM-PM receives the DTP APDU as user data of a P-TOKEN-PLEASE indication primitive, the responding DTAM-PM issues a D-TOKEN-PLEASE indication primitive to the responder. The D-TOKEN-PLEASE indication primitive parameter is derived from the DTP APDU.

6.12.1.3.2 *Token please procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

- a) a D-TOKEN-PLEASE requestor primitive from the requestor;
- b) an S-TOKEN-PLEASE indication primitive.

6.12.1.3.2.1 *D-TOKEN-PLEASE request primitive*

6.12.1.3.2.1.1 If the requesting DTAM-PM does not possess the token and receives a D-TOKEN-PLEASE request from the requestor, DTAM-PM issues a S-TOKEN-PLEASE request primitive. This may be done either inside or outside an activity.

6.12.1.3.2.2 *Implicit DTP APDU*

6.12.1.3.2.2.1 If the responding DTAM-PM receives an S-TOKEN-PLEASE indication primitive without any APDU on its user data, the responding DTAM-PM issues a D-TOKEN-PLEASE indication primitive to the responder.

6.12.1.4 *Use of the DTP APDU fields*

The DTP APDU fields are use as specified below.

6.12.1.4.1 *Priority*

This parameter is the priority of the action, governed by the data token, that the requestor of the D-TOKEN-PLEASE service wishes to carry out. This parameter has to be supplied by the requestor of the D-TOKEN-PLEASE service.

6.12.2 *Token give control*

6.12.2.1 *Purpose*

6.12.2.1.1 The token-give procedure is used by a requestor (sender of documents) to give the token to the responder (receiver of documents).

6.12.2.1.2 The requestor becomes the receiver and the responder becomes the sender.

6.12.2.2 *APDUs used*

No APDUs are used in this procedure.

6.12.2.3 *Token give procedure*

6.12.2.3.1 *Token give procedure mapped onto presentation service (normal mode)*

This procedure is driven by the following events:

- a) a D-TOKEN-GIVE request primitive;
- b) a P-TOKEN-GIVE indication primitive.

6.12.2.3.1.1 *D-TOKEN-GIVE request primitive*

6.12.2.3.1.1.1 If the requesting DTAM-PM possesses the token and receives a D-TOKEN-GIVE request primitive from the requestor, it issues a P-TOKEN-GIVE request primitive and becomes the responding DTAM-PM. This may be done only outside an activity.

6.12.2.3.1.2 *P-TOKEN-GIVE indication primitive*

6.12.2.3.1.2.1 If the responding DTAM-PM receives a P-TOKEN-GIVE indication primitive, the responding DTAM-PM issues a D-TOKEN-GIVE indication primitive to the responder. The responding DTAM-PM becomes the requesting DTAM-PM.

6.12.2.3.2 *Token give procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

- a) a D-TOKEN-GIVE request primitive from the requestor;
- b) an S-TOKEN-GIVE indication primitive.

6.12.2.3.2.1 *D-TOKEN-GIVE request primitive*

6.12.2.3.2.1.1 If the requesting DTAM-PM possesses the token and receives a D-TOKEN-GIVE request primitive from the requestor, it issues an S-TOKEN-GIVE request primitive and becomes the responding DTAM-PM. This may be done only outside an activity.

6.12.2.3.2.2 *S-TOKEN-GIVE indication primitive*

6.12.2.3.2.2.1 If the responding DTAM-PM receives an S-TOKEN-GIVE indication primitive, the responding DTAM-PM issues a D-TOKEN-GIVE indication primitive to the responder. The responding DTAM-PM becomes the requesting DTAM-PM.

6.12.3 *Control-give*

6.12.3.1 *Purpose*

6.12.3.1.1 *The CONTROL-GIVE procedure is used by a requestor to give all the tokens to the responder.*

6.12.3.1.2 *The requestor becomes the receiver and the responder becomes the sender.*

6.12.3.2 *APDUs used*

No APDUs are used in this procedure.

6.12.3.3 *CONTROL-GIVE procedure*

6.12.3.3.1 *CONTROL-GIVE procedure mapped onto presentation service (normal mode)*

This procedure is driven by the following events:

- a) a D-CONTROL-GIVE request primitive;
- b) a P-CONTROL-GIVE indication primitive.

6.12.3.3.1.1 *D-CONTROL-GIVE request primitive*

6.12.3.3.1.1.1 If the requesting DTAM-PM possesses the tokens and receives a D-CONTROL-GIVE request primitive from the requestor, it issues a P-CONTROL-GIVE request primitive and becomes the responding DTAM-PM. This may be done only outside an activity.

6.12.3.3.1.2 *P-CONTROL-GIVE indication primitive*

6.12.3.3.1.2.1 If the responding DTAM-PM receives a P-CONTROL-GIVE indication primitive, the responding DTAM-PM issues a D-CONTROL-GIVE indication primitive to the responder. The responding DTAM-PM becomes the requesting DTAM-PM.

6.12.3.3.2 *CONTROL GIVE procedure mapped onto session service (transparent mode)*

This procedure is driven by the following events:

- a) a D-CONTROL-GIVE request primitive from the requestor;
- b) an S-CONTROL-GIVE indication primitive.

6.12.3.3.2.1 *D-CONTROL-GIVE request primitive*

6.12.3.3.2.1.1 If the requesting DTAM-PM possesses the tokens and receives a D-CONTROL-GIVE request primitive from the requestor, it issues an S-CONTROL-GIVE request primitive and becomes the responding DTAM-PM. This may be done only outside an activity.

6.12.3.3.2.2 *S-CONTROL-GIVE indication primitive*

6.12.3.3.2.2.1 If the responding DTAM-PM receives an S-CONTROL-GIVE indication primitive, the responding DTAM-PM issues a D-CONTROL-GIVE indication primitive to the responder. The responding DTAM-PM becomes the requesting DTAM-PM.

6.13 *Exception report*

6.13.1 *User-exception report*

6.13.1.1 *Purpose*

The user-exception-report procedure is used by the responding DTAM-PM to report an error situation to the requesting DTAM-PM during document bulk transfer.

6.13.1.2 *APDUs used*

No APDUs are used in this procedure.

6.13.1.3 *User-exception report procedure*

6.13.1.3.1 *User-exception-report procedure mapped onto presentation service (normal mode)*

This procedure is driven by the following events:

- a) a responding DTAM-PM problem;
- b) a P-U-EXCEPTION-REPORT indication primitive.

6.13.1.3.1.1 *Receiving DTAM-PM problem*

6.13.1.3.1.1.1 If the responding DTAM-PM detects a problem, it issues a P-U-EXCEPTION-REPORT request primitive. Depending on the severity of the detected error, the value of the Reason parameter of the P-U-EXCEPTION-REPORT request primitive is as follows:

- a) in severe problem situations, the value "receiving ability jeopardized" is used;
- b) in exceptional circumstances, the responding DTAM-PM may have to delete a partially received document information, even though some minor synchronization points have been confirmed. In this case, the value "unrecoverable procedure error" is used;
- c) if the responding DTAM-PM is not willing to complete a transfer procedure (see § 6.6.3), the value "non-specific error" is used;
- d) if the requesting DTAM-PM resumes a transfer procedure already finished by the responding DTAM-PM within an application-association, the value "sequence error" is used;
- e) for all other less severe error situations, the value "local SS-User error" is used.

6.13.1.3.1.2 *P-U-EXCEPTION-REPORT indication primitive*

6.13.1.3.1.2.1 If the requesting DTAM-PM receives a P-U-EXCEPTION-REPORT indication primitive, it performs one of the following procedures depending on the Reason parameter value of the P-U-EXCEPTION-REPORT indication primitive and Reliable transfer modes:

- a) With a value "receiving ability jeopardized",
 - the transfer-abort procedure (see § 6.4.3.1.4) followed by the DTAM provider-abort procedure are performed (mode 1/2).
- b) With a value "unrecoverable procedure error",
 - the transfer-discard procedure (see § 6.6.6) followed by D-TRANSFER service (indication and confirmation) are performed (mode 1),
 - the transfer-discard procedure followed by transfer procedure are performed (mode 2).
- c) With a value "non-specific error",
 - the transfer-discard procedure followed by D-TRANSFER service (indication and confirmation) are performed (mode 1),
 - the transfer-discard procedure followed by transfer procedure are performed (mode 2).
- d) With a value "sequence error",
 - the transfer-discard procedure is performed and the requesting DTAM-PM issues an D-TRANSFER confirm primitive with a result parameter value "document-information- transferred" to the requestor and the transfer procedure is finished (mode 1/2).
- e) With a value "local SS-User error" and at least one confirmed checkpoint in the transfer procedure,
 - the transfer-interrupt procedure (see § 6.6.5) followed by D-TRANSFER service (indication and confirmation) are performed (mode 1),
 - the transfer-interrupt procedure followed by the transfer-resumption procedure (see § 6.6.7) are performed (mode 2).

- f) With a value "local SS-User error" and no checkpoint was confirmed in the transfer procedure,
- the transfer-discard procedure followed by D-TRANSFER service (indication and confirmation) are performed (mode 1),
 - the transfer-discard procedure followed by the transfer procedure are performed (mode 2).

6.13.1.3.1.2.2 Table 16/T.433 summarizes the actions of the requesting DTAM-PM when receiving the P-U-EXCEPTION-REPORT indication primitive.

TABLE 16/T.433
**The actions of the requesting DTAM-PM in
the error detection at the responding DTAM-PM**

Parameter values (P/S-U-EXCEPTION-REPORT indication)	List of procedures in reliable transfer mode 1	List of procedures in reliable transfer mode 2
Receiving ability jeopardized	- transfer-abort - DTAM provider-abort	- transfer-abort - DTAM provider abort
Unrecoverable procedure error	- transfer-discard - D-TRANSFER indication/ confirmation - transfer-procedure by DTAM-user ^{a)}	- transfer-discard - transfer-procedure by DTAM-PM
Non-specific error	- transfer-discard - D-TRANSFER indication/ confirmation - transfer-procedure by DTAM-user ^{a)}	- transfer-discard - transfer-procedure by DTAM-PM
Sequence error	- transfer-discard	- transfer-discard
Local SS-user error (any checkpoint)	- transfer-interrupt - D-TRANSFER indication/ confirmation - transfer-resume by DTAM-user ^{a)}	- transfer-interrupt - transfer-resume by DTAM-PM
Local SS-user error (no checkpoint)	- transfer-discard - D-TRANSFER indication/ confirmation - transfer-procedure by DTAM-user ^{a)}	- transfer-discard - transfer-procedure by DTAM-PM

- a) Transfer-resume procedure and transfer-procedure by DTAM-user may be performed in accordance with the decision of DTAM-user.

6.13.1.3.2 User-exception-report procedure mapped onto session service (transparent mode)

This procedure is driven by the following events:

- a) a responding DTAM-PM problem;
- b) an S-U-EXCEPTION-REPORT indication primitive.

6.13.1.3.2.1 *Receiving DTAM-PM problem*

6.13.1.3.2.1.1 If the responding DTAM-PM detects a problem, it issues a S-U-EXCEPTION-REPORT request primitive. Depending on the severity of the detected error, the value of the Reason parameter of the S-U-EXCEPTION-REPORT request primitive is as follows:

- a) in severe problem situations, the value "receiving ability jeopardized" is used;
- b) in exceptional circumstances, the responding DTAM-PM may have to delete a partially received document information, even though some minor synchronization points have been confirmed. In this case, the value "unrecoverable procedure error" is used;
- c) if the responding DTAM-PM is not willing to complete a transfer procedure, the value "non-specific error" is used;
- d) if the requesting DTAM-PM resumes a transfer procedure already finished by the responding DTAM-PM, the value "sequence error" is used;
- e) for all other less severe error situations, the value "local SS-user error" is used.

6.13.1.3.2.2 *S-U-EXCEPTION-REPORT indication primitive*

6.13.1.3.2.2.1 If the requesting DTAM-PM receives an S-U-EXCEPTION-REPORT indication primitive, it performs one of following procedures depending on the Reason parameter value of the S-U-EXCEPTION-REPORT indication primitive (in the transparent mode, only Reliable transfer mode 1 is available):

- a) with a value "receiving ability jeopardized", the transfer-abort procedure followed by the DTAM provider-abort procedure are performed;
- b) with a value "unrecoverable procedure error", the transfer-discard procedure followed by D-TRANSFER service (indication and confirmation) are performed;
- c) with a value "non-specific error", the transfer-discard procedure followed by D-TRANSFER service (indication and confirmation) are performed;
- d) with a value "sequence error", the transfer-discard procedure is performed and the requesting DTAM-PM issues a D-TRANSFER confirm primitive with a result parameter value of "document-information-transferred" to the requestor and the transfer procedure is finished;
- e) with a value "local SS-user error" and at least one confirmed checkpoint in the transfer procedure, the transfer-interrupt procedure followed by D-TRANSFER service (indication and confirmation) are performed. If no checkpoint was confirmed in the transfer procedure, the transfer-discard procedure followed by D-TRANSFER service (indication and confirmation) are performed.

6.13.1.3.2.2.2 Table 16/T.433 summarizes the actions of the requesting DTAM-PM when receiving the S-U-EXCEPTION-REPORT indication primitive in the column of Reliable transfer mode 1.

6.13.2 *Provider-exception-report*

6.13.2.1 *Purpose*

If the presentation service-provider (normal mode) or the session service-provider (transparent mode) (responding side) detects an unexpected situation during an activity, not covered by other services, a P-P-EXCEPTION-REPORT or S-P-EXCEPTION-REPORT indication primitive is respectively issued to both DTAM-PMs.

6.13.2.2 *APDUs used*

No APDUs are used in this procedure.

6.13.2.3 *Provider-exception-report procedure*

6.13.2.3.1 *Provider-exception-report procedure (normal mode)*

This procedure is driven by the following events:

- a) a P-P-EXCEPTION-REPORT indication primitive.

6.13.2.3.1.1 *P-P-EXCEPTION-REPORT indication primitive*

6.13.2.3.1.1.1 The responding DTAM-PM ignores a P-P-EXCEPTION-REPORT indication primitive.

6.13.2.3.1.1.2 If the requesting DTAM-PM receives a P-P-EXCEPTION-REPORT indication primitive, it may perform one of the following procedures:

- a) if at least one checkpoint was confirmed in the transfer procedure,
 - the transfer-interrupt procedure followed by D-TRANSFER service (indication and confirmation) (Reliable transfer mode 1),
 - the transfer-interrupt procedure followed by the transfer-resumption procedure (Reliable transfer mode 2), or
- b) if no checkpoint was confirmed in the transfer procedure,
 - the transfer-discard procedure followed by D-TRANSFER service (indication and confirmation) (Reliable transfer mode 1),
 - the transfer-discard procedure followed by the transfer-retry procedure (Reliable transfer mode 2), or
- c) the transfer-abort procedure followed by the DTAM provider-abort procedure.

6.13.2.3.2 *Provider-exception-report procedure (transparent mode)*

This procedure is driven by the following event:

- a) an S-P-EXCEPTION-REPORT indication primitive.

6.13.2.3.2.1 *S-P-EXCEPTION-REPORT indication primitive*

6.13.2.3.2.1.1 The responding DTAM-PM ignores an S-P-EXCEPTION-REPORT indication primitive.

6.13.2.3.2.1.2 If the requesting DTAM-PM receives an S-P-EXCEPTION-REPORT indication primitive, it may perform one of the following procedure:

- a) if at least one checkpoint was confirmed in the transfer procedure, the transfer-interrupt procedure followed by the DTRANSFER service (indication and confirmation), or
- b) if no checkpoint was confirmed in the transfer procedure, the transfer-discard procedure followed by the D-TRANSFER service (indication and confirmation), or
- c) the transfer-abort procedure followed by the DTAM provider-abort procedure.

6.14 *Rules for extensibility*

In addition to the procedures stated above, the following applies when processing the APDUs defined in this part of Recommendation T.433:

- a) fields are ignored that are not defined in this part of Recommendation T.433 in DTAM association establishing phase PDUs (DINQ, DINR and DAB); and
- b) for fields defined as having a maximum length in this part of Recommendation T.433, that

7 **Mapping to the lower layer services**

7.1 *Mapping to the OSI lower layer services*

This section defines how a DTAMPM transfers APDUs by means of:

- a) the ACSE services, or
- b) the presentation-services.

Table 17/T.433 lists the overview of ACSE or presentation-service mapping.

TABLE 17/T.433

ACSE and presentation services mapping overview

Functional units	Service primitive	Protocol elements (PDU)	Mapping DTAM PDU to ACSE service/presentation service
Association use control	D-INITIATE req/ind D-TERMINATE rsp/cnf D-TERMINATE req/ind D-U-ABORT rsp/cnf D-U-ABORT req/ind	D-INITIATE-REQ PDU D-INITIATE-RESP PDU D-TERMINATE-REQ PDU D-TERMINATE-RESP PDU D-ABORT PDU	A-ASSOCIATE req/ind A-ASSOCIATE rsp/cnf A-RELEASE req/ind A-RELEASE rsp/cnf A-U-ABORT req/ind
Capability	D-CAPABILITY req/ind D-CAPABILITY rsp/cnf	D-CAPABILITY REQ PDU D-CAPABILITY RESP PDU	P-CAPAB-DATA req/ind P-CAPAB-DATA rsp/cnf
Document bulk transfer	D-TRANSFER req ind cnf	none none none - transfer-interrupt - transfer-discard - transfer-resume	P-ACT-START/RESUME ^{a)} req/ind P-DATA req/ind P-ACT-END/DCD ^{a)} /INT ^{a)} req/ind P-ACT-END/DCD ^{a)} /INT ^{a)} rsp/cnf P-ACT-INT req/ind/rsp/cnf P-ACT-DCD req/ind/rsp/cnf P-ACT-RESUME req/ind
Document unconfirmed manipulation	D-CREATE req/ind D-DELETE req/ind D-MODIFY req/ind D-CALL req/ind D-REBUILD ^{b)} req/ind	D-CREATE PDU D-DELETE PDU D-MODIFY PDU D-CALL PDU D-REBUILD PDU ^{b)}	P-DATA req/ind P-DATA req/ind P-DATA req/ind P-DATA req/ind P-DATA req/ind
Token control	D-TOKEN-GIVE req/ind D-TOKEN-PLS req/ind D-CONTROL-GIVE req/ind	none D-TOKEN-PLS PDU none	P-TOKEN-GIVE req/ind P-TOKEN-PLS req/ind P-CONTROL-GIVE req/ind
Typed data transmission	D-TYPED-DATA req/ind	D-TYPED-DATA PDU	P-TYPED-DATA req/ind
Exception report	D-U-EXCEPTION-REPORT req/ind ^{b)}	D-EXCEPTION-REPORT PDU ^{b)} - user-exception-report - provider-exception-report	P-U-EXCEPTION-REPORT req/ind P-U-EXCEPTION-REPORT req/ind P-P-EXCEPTION-REPORT ind

a) This mapping is only applied in Reliable transfer mode 1.

b) This DTAM service or PDU is for further study.

7.1.1 Mapping on the ACSE services

7.1.1.1 Association-establishment procedure

Association-establishment procedure takes place concurrently with the underlying ACSE association establishment.

7.1.1.1.1 Directly mapped parameters

The following parameters of D-INITIATE service primitives are mapped directly onto the corresponding parameter of the A-ASSOCIATE service primitive:

- application context name
- calling AP title
- calling AP invocation-identifier
- calling AE qualifier
- calling AE invocation-identifier
- called AP title
- called AP invocation-identifier
- called AE qualifier
- called AE invocation-identifier
- responding AP title

- k) responding AP invocation-identifier
- l) responding AE qualifier
- m) responding AE invocation-identifier
- n) calling presentation address
- o) called presentation address
- p) responding presentation address
- q) presentation context definition list
- r) presentation context definition result
- s) presentation requirements
- t) initial assignment of token
- u) quality of services.

7.1.1.1.2 *Parameters not used*

The following parameter of AASSOCIATE service primitives is not used:

- Initial synchronization point serial number.

7.1.1.1.3 *Use of the other A-ASSOCIATE request and indication primitive parameters*

7.1.1.1.3.1 *Mode*

This parameter shall be supplied by the requestor of the association in the A-ASSOCIATE request primitive, and shall have the value "normal mode".

7.1.1.1.3.2 *User information*

For both the A-ASSOCIATE request and indication primitives, the user information parameter is used to carry the D-INITIATE-REQ APDU.

7.1.1.1.3.3 *Session requirements*

This parameter is set by the association-initiating DTAM-PM to select the following functional units:

- a) kernel
- b) half-duplex functional unit
- c) typed-data functional unit
- d) capability data exchange functional unit
- e) minor synchronize functional unit
- f) exceptions functional unit
- g) activity management functional unit.

Note – The use of duplex functional unit and negotiated release functional unit are for further study.

7.1.1.1.3.4 *Session connection identifier*

The association-initiating DTAM-PM will supply a session connection identifier, which will be used to uniquely identify the session-connection. This identifier is formed of the following components:

- a) SS-user reference;
- b) common reference;
- c) additional reference information (optionally).

The SS-user reference is conveyed as the calling SS-user reference by the association- initiating DTAM-PM. Common reference and additional reference information are conveyed in similarly named parameters of the P-CONNECT primitive.

Each component, when present, will contain a data element of the appropriate type from the following definitions:

CallingSSUserReference ::= PresentationAddress -- of the requestor

CommonReference ::= UTCTime

AdditionalReferenceInformation ::= T.61 String

The PresentationAddress is represented as a string of octets.

7.1.1.1.4 *Use of the other A-ASSOCIATE response and confirm primitive parameters*

7.1.1.1.4.1 *User information*

This parameter only has relevance if the application-association is accepted by the ACSE service-provider.

For both the A-ASSOCIATE response and confirmation primitives, the user information parameter is used to carry the D-INITIATE-RESP APDU, whether the application-association is accepted or is rejected by the association-responding DTAM-PM.

7.1.1.1.4.2 *Result*

For the A-ASSOCIATE response primitive the result parameter is set by the association responding DTAM-PM as follows:

- a) if the association-responding DTAM-PM rejects the application-association, the value of this parameter is set to either "rejected by responder (transient)" or "rejected by responder (permanent)";
- b) if the association-responding DTAM-PM accepts the request, the value of this parameter is derived from the result parameter of the D-INITIATE response primitive.

7.1.1.1.4.3 *Session requirements*

This parameter has the same values as in the A-ASSOCIATE request and indication primitive.

7.1.1.1.4.4 *Session connection identifier*

This parameter has the same value as in the A-ASSOCIATE indication primitives. The calling SS-user reference value of the A-ASSOCIATE indication primitive is returned as a called SS=user reference by the association-responding DTAM-PM.

7.1.1.2 *Association-release procedure*

Association-release procedure takes place concurrently with the underlying ACSE association release.

7.1.1.2.1 *Directly mapped parameters*

The following parameter of D-TERMINATE service primitives is mapped directly onto the corresponding parameters of the A-RELEASE service primitives:

- User data (on user information).

7.1.1.2.2 *Parameters not used*

The following parameter of the A-RELEASE service primitives is not used:

- Reason.

7.1.1.2.3 *Use of the other A-RELEASE response and confirm primitive parameters*

7.1.1.2.3.1 *Result*

The value of this parameter is "affirmative".

7.1.1.3 *Association-provider-abort*

The use of the A-P-ABORT indication primitive parameters are defined in Recommendation X.217.

7.1.2 *Mapping on the presentation services*

7.1.2.1 *Transfer procedure*

7.1.2.1.1 *Use of the P-ACTIVITY-START request and indication primitive parameters*

7.1.2.1.1.1 *Activity identifier*

The activity identifier identifies the activity by means of a serial number. The first activity started on the session-connection is assigned to number 1. Each successive activity for that direction of transfer is assigned the next number. This number is separate for each direction of transfer.

The DTAM-PMs should manage the local mapping between the parameter "activity identifier" in the P-ACTIVITY-START request and indication primitive and the parameter "document reference information in D-TRANSFER service.

The property required of activity identifiers is that they should uniquely identify an activity during a reasonable time interval within a particular session-connection, so that duplicates can be detected in the case of error situations. These identifiers are allocated by numbering the activities during a session, starting with one for the first and incrementing for each successive activity, and representing the number by a data element of type INTEGER encoded according to Recommendation X.209. It is unnecessary for the responding DTAM-PM to make assumptions on allocation method, it need only be able to compare two identifiers for equality, octet by octet.

7.1.2.1.1.2 *User data*

This parameter is not used.

7.1.2.1.2 *Use of the P-DATA request and indication primitive parameters*

7.1.2.1.2.1 *User data*

The following DTAM APDUs are conveyed by this parameter:

- a) D-CREATE APDU
- b) D-DELETE APDU
- c) DMODIFY APDU
- d) D-CALL APDU.

Note – A segment of interchange-data-elements of document information is also conveyed by this parameter.

7.1.2.1.3 *Use of the P-TYPED-DATA request and indication primitive parameters*

7.1.2.1.3.1 *User data*

The D-TYPED-DATA APDU is conveyed by this parameter.

7.1.2.1.4 *Use of the P-MINOR-SYNCHRONIZE request and indication primitive parameters*

7.1.2.1.4.1 *Type*

The DTAM-PM uses only the "explicit confirmation expected" type of minor synchronization.

7.1.2.1.4.2 *Synchronization point serial number*

The session service-provider allocates checkpoint serial numbers, and passes then to the requesting and responding DTAM-PMs to associate with the transmitted data.

7.1.2.1.4.3 *User data*

This parameter is not used.

7.1.2.1.5 *Use of the P-ACTIVITY-END request and indication primitive parameters*

7.1.2.1.5.1 *User data*

This parameter is not used.

7.1.2.2 *Token please procedure*

When the DTAM user issues a D-TOKEN-PLEASE service primitive, this results into a P-TOKEN-PLEASE.

7.1.2.2.1 *Use of the P-TOKEN-PLEASE request and indication primitive parameters*

7.1.2.2.1.1 *Tokens*

This parameter takes a value which corresponds to requesting the data token.

7.1.2.2.1.2 *User data*

The D-TOKEN-PLEASE APDU is conveyed by this parameter.

7.1.2.3 *Token give procedure*

When the DTAM user issues a D-TOKEN-GIVE service primitive, this results into a P-TOKEN-GIVE.

7.1.2.3.1 *Use of the P-TOKEN-GIVE request and indication primitive parameters*

7.1.1.3.1.1 *Tokens*

This parameter takes a value which corresponds to giving the data token.

7.1.2.4 *Control give procedure*

When the DTAM-user issues a D-CONTROL-GIVE service primitive, this results into a P-CONTROL-GIVE. This will transfer all the tokens from the requestor to the responder.

7.1.2.4.1 *Use of the P-CONTROL-GIVE request and indication primitive parameters*

The P-CONTROL-GIVE service primitives have no parameters. All the tokens are automatically passed to the other DTAMPM.

7.1.2.5 *Capability data exchange procedure*

7.1.2.5.1 *Use of the P-CAPAB-DATA service parameters*

7.1.2.5.1.1 *User data*

The following DTAM APDUs are conveyed by this parameter:

- a) D-CAPABILITY-REQ APDU;
- b) D-CAPABILITY-RESP APDU.

7.1.2.6 *User-exception-report procedure*

7.1.2.6.1 *Use of the P-U-EXCEPTION-REPORT service parameters*

7.1.2.6.1.1 *Reason*

This parameter may specify one of the following reasons:

- a) receiving ability jeopardized;
- b) local SS-user error;
- c) sequence error;
- d) unrecoverable error;
- e) non-specific error.

7.1.2.6.1.2 *User data*

This parameter is not used.

7.1.2.7 *Provider-exception-report procedure*

7.1.2.7.1 *Use of the P-P-EXCEPTION-REPORT service parameter*

7.1.2.7.1.1 *Reason*

This parameter may specify one of the following reasons:

- a) protocol error,
- b) non-specific error.

7.1.2.8 *Transfer-interrupt procedure*

7.1.2.8.1 *Use of the P-ACTIVITY-INTERRUPT service parameters*

7.1.2.8.1.1 *Reason*

This parameter may specify one of the following reasons:

- a) local SS-user error;
- b) non-specific error.

7.1.2.9 *Transfer-discard procedure*

7.1.2.9.1 *Use of the P-ACTIVITY-DISCARD service parameters*

7.1.2.9.1.1 *Reason*

This parameter may specify one of the following reasons:

- a) local SS-user error;
- b) unrecoverable procedure error;
- c) non-specific error.

7.1.2.10 *Transfer-resumption procedure*

7.1.2.10.1 *Use of the P-ACTIVITY-RESUME service parameters*

7.1.2.10.1.1 *Activity identifier*

The requesting DTAM-PM must allocate and supply the next activity identifier number for the current session.

7.1.2.10.1.2 *Old activity identifier*

The requesting DTAM must supply the original activity identifier assigned to the previously interrupted activity in the P-ACTIVITY-START request primitive.

7.1.2.10.1.3 *Synchronization point serial number*

The requesting DTAM-PM will specify the serial number of the last confirmed checkpoint in the interrupted activity. The session service-provider will also set the current session serial number to this value. If there was no previously confirmed checkpoint, the activity cannot be continued. The requesting DTAM-PM must then send a P-ACTIVITY-RESUME request primitive (with the synchronization point serial number set to zero), followed by a P-ACTIVITY-DISCARD request primitive.

7.1.2.10.1.4 *Old session connection identifier*

The requesting DTAM-PM must supply the session connection identifier of the session-connection during which the activity was started. The session connection identifier of the previous session-connection is conveyed in the calling SS-user reference, common reference and optionally additional reference information components of this parameter. The called SS-user reference component is not used.

7.1.2.10.1.5 *User data*

This parameter is not used.

7.2 *Mapping to the Recommendation X.215 session service (transparent mode)*

This section defines how a DTAM-PM transfers APDUs by means of the session service.

Table 20/T.433 lists the overview of session mapping.

7.2.1 DTAM association-establishment procedure

The association-establishment procedure takes place concurrently with the underlying establishment of session connection.

7.2.1.1 Directly mapped parameters

No parameters of D-INITIATE service primitives are mapped directly onto the corresponding parameters of the S-CONNECT service primitive.

7.2.1.2 Use of the other S-CONNECT request and indication primitive parameters

7.2.1.2.1 User information

For both the S-CONNECT request and indication primitives, the user information parameter is used to carry the D-INITIATE-REQ APDU.

7.2.1.2.2 Session requirements

This parameter is set by the initiating DTAM-PM to select the following functional units by means of the "telematic requirements parameter" in the D-INITIATE service primitive as shown in Table 18/T.433.

TABLE 18/T.433

Mapping into/out of the session requirements

"Telematic requirements"	Functional units
Token management	Half-duplex functional unit
Non-token management	Duplex functional unit
Typed data management	Typed-data functional unit
Capability	Capability data exchange functional unit
Reliable transfer management	Minor synchronize functional unit Activity management functional unit
Exception report	Exceptions functional unit

7.2.1.2.3 Session reference

The initiating DTAM-PM will supply a session connection identifier, which will be used to uniquely identify the session-connection. This identifier is formed of the following components:

- terminal identifier of the calling terminal;

- b) date and time;
- c) additional session reference number (optionally).

The terminal identifier of the calling terminal is conveyed as the calling SS-user reference by the initiating DTAM-PM. Date and time and additional session reference number are conveyed in parameters of the S-CONNECT primitive.

Each component, when present, will contain a data element of the appropriate type from the following definitions:

Terminal identifier	::= T.62 SessionAddress -- of the requestor
Date and time	::= UTCTime
Additional session reference number	::= T.61 String

The SessionAddress is represented as a string of octets.

7.2.1.2.4 *Service identifier*

The initiating DTAM-PM must supply a service identifier which has the value '1' to specify the telematic services.

7.2.1.2.5 *Non-basic session capabilities*

The initiating DTAM-PM may supply non-basic session capabilities, which will be used to specify the non-basic session capabilities available as receiving capabilities of the sender of this primitive. This parameter is formed of the following components:

- a) miscellaneous session capabilities;
- b) window size.

7.2.1.2.6 *Inactivity timer*

The initiating DTAM-PM may use to negotiate the value of an inactivity timer.

7.2.1.3 *Use of the other S-CONNECT response and confirm primitive parameters*

7.2.1.3.1 *User information*

This parameter only has relevance if the application-association is accepted by the session service-provider.

For both the S-CONNECT response and confirmation primitives, the user information parameter is used to carry the D-INITIATE-RESP APDU if the application-association is accepted or is rejected by the responding DTAM-PM.

7.2.1.3.2 *Result*

For the S-CONNECT response primitive the result parameter is set by the association responding DTAM-PM as follows:

- a) when the association-responding DTAM-PM detects errors in the S-CONNECT indication primitive (e.g., no session user data), the association-responding DTAM-PM rejects the application-association. The value of this parameter is set to "refuse";
- b) if the association-responding DTAM-PM accepts the request, the value of this parameter is derived from the result parameter of the D-INITIATE response primitive as shown in Table 19/T.433.

For the S-CONNECT confirm primitive the result parameter is set by the association responding DTAM-PM as follows:

- a) when the association-requesting (initiating) DTAM-PM receives the S-CONNECT confirm primitive with the result parameter of "refuse", the association/requesting DTAM-PM issues the D-INITIATE confirm primitive with the result parameter of "rejected by responding DTAM-PM" to the association-requesting DTAM user;
- b) when the association-requesting DTAM-PM receives the S-CONNECT confirm primitive with the result parameter of "accept", the association-requesting DTAM-PM issues the D-INITIATE confirm primitive with the result parameter of "accepted" to the association-requesting DTAM user.

TABLE 19/T.433

Result parameter mapping

D-INITIATE response	S-CONNECT response/ confirmation	D-INITIATE confirmation
accepted	accept	accepted
rejected by responder with some reasons	refuse	rejected by responding DTAM-PM
rejected by responding DTAM-PM	refuse	rejected by responding DTAM-PM

TABLE 20/T.433

DTAM service primitive-protocol mapping to session services

Functional units	Service primitive	Protocol elements (PDU)	Mapping DTAM PDU to session service
Association use control	D-INITIATE req/ind resp/conf D-TERMINATE req/ind resp/conf D-ABORT req/ind	D-INITIATE-REQ PDU D-INITIATE-RESP PDU	S-CONNECT req/ind S-CONNECT rsp/cnf S-RELEASE req/ind S-RELEASE rsp/cnf S-ABORT req/ind
Capability	D-CAPABILITY req/ind resp/conf	D-CAPABILITY-REQ PDU D-CAPABILITY-RESP PDU	S-CAPAB-DATA req/ind S-CAPAB-DATA rsp/cnf
Document bulk transfer	D-TRANSFER req ind cnf	none none none	S-ACT-START/RESUME req/ind S-DATA req/ind S-ACT-END/DCD/INT req/ind S-ACT-END/DCD/INT rsp/cnf
Token control	D-CONTROL-GIVE req/ind D-TOKEN-PLS req/ind	none none	S-CONTROL-GIVE req/ind S-TOKEN-PLS req/ind
Typed data transmission	D-TYPED-DATA req/ind	D-TYPED-DATA PDU	S-TYPED-DATA req/ind
Exception report	D-U-EXCEPTION-REPORT req/ind*	- user-exception-report - provider-exception-report	S-U-EXCEPTION-REPORT req/ind S-U-EXCEPTION-REPORT req/ind S-P-EXCEPTION-REPORT ind

Note – D-U-EXCEPTION-REPORT req/ind* is for further study.

7.2.1.3.3 Session requirements

This parameter has the same values as in the S-CONNECT request and indication primitives.

7.2.1.3.4 Session reference

This parameter has the same values as in the S-CONNECT indication primitives. The terminal identifier of the calling terminal value of the S-CONNECT indication primitive is returned as the terminal identifier of the called terminal by the responding DTAM-PM.

7.2.1.3.5 Service identifier

This parameter has the same values as in the S-CONNECT request and indication primitives.

7.2.1.3.6 *Non-basic session capabilities*

The responding DTAM-PM may supply non-basic session capabilities, which will be used to specify the non basic session capabilities available as receiving capabilities of the sender of this primitive. This parameter is formed of the same components as those in S-CONNECT request and indication primitives.

7.2.1.3.7 *Inactivity timer*

The responding DTAM-PM may use this parameter to negotiate an inactivity timer.

7.2.2 *Association release procedure*

The association release procedure takes place concurrently with the underlying release of session connection.

There are no D-TERMINATE service parameters to map onto session connection release service parameters.

7.2.3 *Association-provider-abort*

The uses of the S-P-ABORT indication primitive parameters are defined in Recommendation X.215.

7.2.4 *Transfer procedure*

7.2.4.1 *Use of the S-ACTIVITY-START request and indication primitive parameters*

7.2.4.1.1 *Document reference number*

The requesting DTAM-PM must allocate and supply the next document reference number for the current session.

The DTAM-PMs should manage the mapping between the parameter "document reference number" in D-TRANSFER service and the parameter "document reference number" in the S-ACTIVITY-START request and indication primitives.

7.2.4.1.2 *Document type identifier*

This parameter may be user option.

7.2.4.1.3 *Service interworking identifier*

This parameter may be user option.

7.2.4.1.4 *User data*

This parameter is only used to invoke the DTAM capability. The information, which is generated by the DTAM-PM based on the parameter of "document characteristics" in document profile contained in the document information, is conveyed as shown in Figure 3/T.433.

```

S-ACTIVITY-START-user-data ::= CHOICE

    (      [4] IMPLICIT DocumentCharacteristics      )

DocumentCharacteristics ::= SET(
    documentApplicationProfile ::= CHOICE (
        [0] IMPLICIT OCTET STRING
        -- '01'H Non-DocumntApplicationProfile
        -- '02'H DocumentApplicationProfile T.503
        [4] IMPLICIT SET OF OBJECT IDENTIFIER )

    documentArchitectureClass      [1] IMPLICIT OCTET STRING OPTIONAL,
        -- '00'H means FDA

    nonBasicDocumentCharacteristics [2] IMPLICIT NonBasicDocumentCharacteristics
        OPTIONAL,

    nonBasicStructuralCharacteristics [3] IMPLICIT NonBasicStructuralCharacteristics
        OPTIONAL )

NonBasicDocumentCharacteristics ::= SET (
    commentsCharacterSets      [1] IMPLICIT OCTET STRING OPTIONAL,
        -- string of escape sequences
    pageDimensions             [2] IMPLICIT SET OF Dimension-Pair OPTIONAL,
    ra-gr-coding-attributes     [3] IMPLICIT SET OF Ra-Gr-Coding-Attribute OPTIONAL,
        -- Ra-Gr-Coding-Attribute is defined in
        -- Recommendation T.415

    ra-gr-presentation-features [4] IMPLICIT SET OF
        Ra-Gr-Presentation-Features OPTIONAL
        -- Ra-Gr-Presentation-Features is defined in
        -- Recommendation T.415 )

NonBasicStructuralCharacteristics ::= SET(
    numberOfObjectsPerPage      [0] IMPLICIT INTEGER OPTIONAL)

```

FIGURE 3/T.433

User data in S-ACTIVITY-START/RESUME

7.2.4.2 Use of the S-DATA request and indication primitive parameters

The document information is divided into segments such that the segment boundaries coincide with the minor synchronization points. Each segment consists of an integral number of interchange- data-elements. The interchange- data-elements of each segment are encoded using the basic encoding rules defined in Recommendation X.209. The encoded interchange- data-elements of each segment are concatenated, forming an encoded segment.

7.2.4.2.1 User data

A segment of interchange- data-elements is conveyed by the user data.

Note – Some DTAM-PMs may take one of the following actions:

- when sending the document information, the requesting DTAM-PM may suppress the document profile located at the top of the document information;
- when receiving the document information, the responding DTAM-PM may re-generate the document profile and may attach it to the top of the document information based on the User Data of S-ACTIVITY-START indication.

7.2.4.3 Use of the S-TYPED-DATA request and indication primitive parameters

7.2.4.3.1 User data

The D-TYPED-DATA APDU is conveyed by this parameter.

7.2.4.4 *Use of the S-MINOR-SYNCHRONIZE service parameters*

7.2.4.4.1 *Type*

The DTAM-PM uses only the "explicit confirmation expected" type of minor synchronization.

7.2.4.4.2 *Synchronization point serial number (checkpoint reference number)*

The session service-provider allocates checkpoint serial numbers and passes them to the requesting and the responding DTAM-PMs to associate with the transmitted data.

7.2.4.5 *Use of the S-ACTIVITY-END service parameters*

7.2.4.5.1 *Synchronization point serial number (checkpoint reference number)*

The serial number of the implied major synchronization point is allocated by the session service-provider and passed up to both DTAM-PMs. This parameter will be mapped into/out of the parameter "synchronization point" in D-TRANSFER service.

7.2.5 *Token please procedure*

When the DTAM-user issues a D-TOKEN-PLEASE service primitive, this results into an S-TOKEN-PLEASE.

7.2.5.1 *Use of the S-TOKEN-PLEASE request and indication primitive parameters*

7.2.5.1.1 *Tokens*

The responding DTAM-PM (receiver of document) will only request the data token.

7.2.6 *Control give procedure*

When the DTAM-user issues a D-CONTROL-GIVE service primitive, this results into an S-CONTROL-GIVE. This will transfer all the tokens from the requestor to the responder.

7.2.6.1 *Use of the S-CONTROL-GIVE request and indication primitive parameters*

The S-CONTROL-GIVE service primitives have no parameters.

7.2.7 *Capability data exchange procedure*

7.2.7.1 *Use of the S-CAPAB-DATA service parameters*

7.2.7.1.1 *User data*

The following DTAM APDUs are conveyed by this parameter:

- a) D-CAPABILITY-REQ APDU;
- b) D-CAPABILITY-RESP APDU.

7.2.7.1.2 *Inactivity timer*

The initiating/responding DTAM-PMs may use this parameter to negotiate an inactivity timer.

7.2.7.1.3 *Storage capacity*

The initiating/responding DTAM-PMs may supply a storage capacity to negotiate the memory size for the communication.

7.2.8 *User-exception-report procedure*

7.2.8.1 *Use of the S-U-EXCEPTION-REPORT service parameters*

7.2.8.1.1 *Reason*

This parameter may specify one of the following reasons:

- a) no specific reason;
- b) temporarily unable to enter into, or to continue a session;

- c) sequence error;
- d) unrecoverable error;
- e) local terminal error.

7.2.9 *Provider-exception-report procedure*

7.2.9.1 *Use of the S-P-EXCEPTION-REPORT service parameters*

7.2.9.1.1 *Reason*

This parameter may specify one of the following reasons:

- protocol error.

7.2.10 *Transfer-interrupt procedure*

7.2.10.1 *Use of the S-ACTIVITY-INTERRUPT service parameters*

7.2.10.1.1 *Reason*

This parameter may specify one of the reasons as described in reason for S-U-EXCEPTION-REPORT service primitive.

7.2.11 *Transfer-discard procedure*

7.2.11.1 *Use of the S-ACTIVITY-DISCARD service parameters*

7.2.11.2.1 *Reason*

This parameter may specify one of the reasons as described in reason for S-U-EXCEPTION-REPORT service primitive.

7.2.12 *Transfer-user-resumption procedure*

7.2.12.1 *Use of the S-ACTIVITY-RESUME service parameters*

7.2.12.1.1 *Document reference number*

The requesting DTAM-PM must allocate and supply the next document reference number for the current session.

7.2.12.1.2 *Old document reference number*

The requesting DTAM-PM must supply the original activity identifier assigned to the previously interrupted activity in the S-ACTIVITY-START request primitive.

7.2.12.1.3 *Checkpoint serial number*

The requesting DTAM-PM will specify the serial number of the last confirmed checkpoint in the interrupted activity. The session service-provider will also set the current session serial number to this value. If there was no previously confirmed checkpoint, the activity cannot be continued. The requesting DTAM-PM must then send an S-ACTIVITY-RESUME request primitive (with the synchronization point serial number set to zero), followed by an S-ACTIVITY-DISCARD request primitive.

This parameter will be mapped into/out of the parameter "synchronization point" in D-TRANSFER service.

7.2.12.1.4 *Old session reference*

The requesting DTAM-PM must supply the session reference of the session-connection during which the activity was started. The session reference of the previous session-connection is conveyed in the calling and called terminal identifier, common reference and optionally, additional reference information components of this parameter.

7.2.12.1.5 *Document type identifier*

This parameter may be user option.

7.2.12.1.6 Service interworking identifier

This parameter may be user option.

7.2.12.1.7 User data

This parameter has the same format of user-data as for the S-ACTIVITY-START service parameters.

8 Abstract syntax definition of APDUs

8.1 Abstract syntax definition of APDUs in normal mode

This abstract syntax is described by the notation of ASN.1 defined in the Recommendation X.208.

DTAM-APDUs {ccitt dTAM(x) apdus(0)} DEFINITIONS ::=

BEGIN

EXPORTS dTAMSE;

dTAMSE OBJECT IDENTIFIER ::= {ccitt dTAM(x) aseID(1)} -- ASE identifier for DTAMSE

IMPORTS

-- For further study

```
[1] D-INITIATE-REQ ::= [APPLICATION 10] IMPLICIT SEQUENCE
{
    serviceClasses [0] IMPLICIT INTEGER OPTIONAL
        -- the use of this parameter is
        -- for further study
    telematicRequirements [1] IMPLICIT BIT STRING
        {
            kernel (0),
            capabilityManagement (1),
            documentBulkTransfer (2),
            typedDataTransmission (3),
            documentUnconfirmedManipulation (4),
            documentConfirmedManipulation (5),
            remoteDocumentAccess (6),
            remoteDocumentManagement (7),
            tokenControl (8),
            exceptionReport (9),
            reliableTransferMode1 (10),
            reliableTransferMode2 (11), },
    applicationCapabilities [2] IMPLICIT SET OF Application Capabilities
    protocolVersion [3] IMPLICIT BIT STRING
        { version-1 (0) } OPTIONAL,
    dTAMQOS [4] IMPLICIT BIT STRING OPTIONAL,
    account [5] IMPLICIT Account OPTIONAL,
    checkpointWindow [6] IMPLICIT INTEGER DEFAULT 3 OPTIONAL,
    storageCapacity [7] IMPLICIT BIT STRING OPTIONAL,
    userInformation [8] OCTET STRING OPTIONAL
}
```

-- Registration of OBJECT IDENTIFIER for the Recommendation T.503 is required

-- dTAMQOS and account parameters are for further study

```
Application Capabilities ::= SET
{
    documentApplicationProfile [0] IMPLICIT OBJECT IDENTIFIER,
    nonBasicDocCharacteristics [1] IMPLICIT NonBasicDocCharacteristics OPTIONAL,
    nonBasicStrucCharacteristics [2] IMPLICIT NonBasicStrucCharacteristics, OPTIONAL,
    operationalApplicationProfile [3] IMPLICIT SET OF OBJECT IDENTIFIER OPTIONAL }
```

-- Registration of OBJECT IDENTIFIER for the Recommendation T.503 is required

-- NonBasicStrucCharacteristics ::= {Refer to Recommendation T.415 }

-- NonBasicDocCharacteristics ::= {Refer to Recommendation T.415 }

```
[2] D-INITIATE-RESP ::= [APPLICATION 11] IMPLICIT SEQUENCE
```


{ telematicRequirements ApplicationCapabilities protocolVersion dTAMQOS result { accepted rejected by responder(reason-not-specified) rejected by responder(protocolVersion-not-supported) rejected by responder(DTAMQOS-not-supported) rejected by responder(application-context-not-supported) rejected by responding DTAM-PM checkpointWindow storageCapacity userInformation }	[0]	IMPLICIT BIT STRING,	
	[1]	IMPLICIT SET OF Application Capabilities,	
	[2]	IMPLICIT BIT STRING { version-1 (0) }	OPTIONAL,
	[3]	IMPLICIT BIT STRING	OPTIONAL,
	[4]	INTEGER	
			(0),
			(1),
			(2),
			(3),
			(4),
			(5) },
	[5]	IMPLICIT INTEGER DEFAULT 3	OPTIONAL,
	[6]	IMPLICIT BIT STRING	OPTIONAL,
	[7]	OCTET STRING	OPTIONAL
[3] D-TERMINATE-REQ	::=	IMPLICIT SEQUENCE	
{ userInformation	[0]	OCTET STRING	OPTIONAL
}			
[4] D-TERMINATE-RESP	::=	IMPLICIT SEQUENCE	
{ charging	[0]	IMPLICIT Charging	OPTIONAL,
userInformation	[1]	OCTET STRING	OPTIONAL
}			
[5] D-ABORT-REQ	::=	[APPLICATION 13] IMPLICIT SEQUENCE	
{ aBORTSource	[0]	INTEGER	
	{	requestingDTAMPM	(0),
		DTAMserviceProvider	(1) },
aBORTReason	[1]	INTEGER	
	{	local-system-problem	(0),
		invalid-parameter	(1),
		unrecognized-activity	(2),
		temporary-problem	(3),
		protocol-error	(4),
		permanent-error	(5),
		transfer-completed	(6) },
Reflected-parameter	[2]	IMPLICIT BIT STRING	OPTIONAL,
-- 8 bits maximum, only if abortReason is invalid parameter			
userInformation	[3]	OCTET STRING	OPTIONAL
}			
[6] D-CAPABILITY-REQ	::=	[APPLICATION 14] IMPLICIT SEQUENCE	
{ applicationCapabilities	[0]	IMPLICIT Application Capabilities	OPTIONAL,
storageCapacity	[1]	IMPLICIT BIT STRING	OPTIONAL,
userInformation	[2]	OCTET STRING	OPTIONAL
}			
[7] D-CAPABILITY-RESP	::=	[APPLICATION 15] IMPLICIT SEQUENCE	
{ applicationCapabilities	[0]	IMPLICIT Application Capabilities	OPTIONAL,
storageCapacity	[1]	IMPLICIT BIT STRING	OPTIONAL,
capabilityResult	[2]	IMPLICIT Capability Result	
userInformation	[3]	OCTET STRING	OPTIONAL
}			
CapabilityResult	::=	INTEGER	
	{	confirmation-of-all-the-	
		requestedCapabilities	(0),
		a-list-of-the-requestedCapabilities	(1),
		a-complete-list-of-non-	
		basic ReceivingCapabilities	(2),

		none-of-the-capabilities-requested-by-the-initiator	(3) }
[8] D-TYPED-DATA	::=	[APPLICATION 16] CHOICE	
	{	NumericString, PrintableString, TeletexString, VideotexString, VisibleString, OctetString, IA5String, GraphicString }	
[9] D-CREATE	::=	[APPLICATION 17] IMPLICIT SEQUENCE OF CreateInformation	
CreateInformation	::=	SEQUENCE	
	{	[0] IMPLICIT ParentObjectOrClassIdentifier OPTIONAL, [1] Object }	
ParentObjectOrClassIdentifier	::=	ObjectOrClassIdentifier	
-- ObjectOrClassIdentifier ::=		{ Refer to Recommendation T.415 }	
Objet	::=	CHOICE	
	{	[0] IMPLICIT DocumentProfileDescriptor, [1] IMPLICIT LayoutClassDescriptor, [2] IMPLICIT LayoutObjectDescriptor, [3] IMPLICIT TextUnit, [5] IMPLICIT LogicalClassDescriptor, [6] IMPLICIT LogicalObjectDescriptor, [7] IMPLICIT PresentationStyleDescriptor, [8] IMPLICIT LayoutStyleDescriptor, <i>-- The above descriptors and text portion are defined -- in Recommendation T.415</i> [9] IMPLICIT OperationalDescriptor. <i>-- The above descriptor is defined -- in Recommendations T.441 and T.541 }</i>	
[10] D-DELETE	::=	[APPLICATION 18] IMPLICIT DeleteInformation	
DeleteInformation	::=	SEQUENCE OF CHOICE	
	{	[0] IMPLICIT ObjectOrClassIdentifier, [1] IMPLICIT ContentPortionIdentifier, <i>-- The above descriptors and text unit are defined -- in Recommendation T.415</i> [2] IMPLICIT OperationalInformationIdentifier <i>-- The above identifiers are defined -- in Recommendations T.441 and T.541 }</i>	
[11] D-MODIFY	::=	[APPLICATION 19] IMPLICIT SEQUENCE OF ModifyInformation	
ModifyInformation	::=	SEQUENCE	
	{	[0] IMPLICIT CurrentObjectOrClassIdentifier OPTIONAL, [1] Object }	
CurrentObjectOrClassIdentifier	::=	ObjectOrClassIdentifier	
-- ObjectOrClassIdentifier	::=	{ refer to Recommendation T.415 }	
Object	::=	CHOICE	
	{	[0] IMPLICIT DocumentProfileDescriptor, [1] IMPLICIT LayoutClassDescriptor, [2] IMPLICIT LayoutObjectDescriptor, [3] IMPLICIT TextUnit,	

```

[5]      IMPLICIT LogicalClassDescriptor,
[6]      IMPLICIT LogicalObjectDescriptor,
[7]      IMPLICIT PresentationStyleDescriptor,
[8]      IMPLICIT LayoutStyleDescriptor,

-- The above descriptors and text unit are defined
-- in Recommendation T.415
[9]      IMPLICIT OperationalDescriptor,
-- The above descriptor is defined
-- in Recommendations T.441 and T.541 }

[12] D-CALL ::= [APPLICATION 20] IMPLICIT CALLInformation
CALLInformation ::= SEQUENCE OF CHOICE
{
[0]      IMPLICIT OperationalInformationIdentifier
-- The above descriptors and text unit are defined
-- in Recommendations T.441 and T.541 }

[13] D-REBUILD ::= [APPLICATION 21] IMPLICIT REBUILDInformation
[for further study]

[14] D-TOKEN-PLEASE ::= [APPLICATION 22] IMPLICIT Priority
Priority ::= INTEGER

END -- of DTAM Protocol Normal Mode

```

8.2 Abstract syntax definition of APDUs for use of session service

This abstract syntax is described by the notation of ASN.1 defined in the Recommendation X.208.

[1] D-INITIATE-REQ ::= CHOICE

```

{
[4]      IMPLICIT ApplicationCapabilities }

ApplicationCapabilities ::= SET {
documentApplicationProfileT73
[0]      IMPLICIT OCTET STRING OPTIONAL,
-- '02'H document application profile (T.503)
documentArchitectureClass
[1]      IMPLICIT OCTET STRING OPTIONAL
-- '00'H means FDA }

```

[2] D-INITIATE-RESP ::= CHOICE

```

{
[4]      IMPLICIT ApplicationCapabilities }

ApplicationCapabilities ::= SET {
documentApplicationProfileT73
[0]      IMPLICIT OCTET STRING OPTIONAL,
-- '02'H document application profile (T.503)
documentArchitectureClass
[1]      IMPLICIT OCTET STRING OPTIONAL
-- '00'H means FDA }

```

[3] D-CAPABILITY-REQ ::= CHOICE

```

{
[4]      IMPLICIT ApplicationCapabilities }

ApplicationCapabilities ::= SET {
documentApplicationProfileT73
[0]      IMPLICIT OCTET STRING OPTIONAL,
documentArchitectureClass
[1]      IMPLICIT OCTET STRING OPTIONAL,
nonBasicDocCharacteristics
[2]      IMPLICIT NonBasicDocCharacteristics OPTIONAL,
nonBasicStrucCharacteristics
[3]      IMPLICIT NonBasicStrucCharacteristics OPTIONAL }

-- "NonBasicDocCharacteristics" and "NonBasicStrucCharacteristics" are defined
-- in Recommendation T.415

```

[4] D-CAPABILITY-RESP ::= CHOICE

```

{
[4]      IMPLICIT ApplicationCapabilities }

ApplicationCapabilities ::= SET {
documentApplicationProfileT73
[0]      IMPLICIT OCTET STRING OPTIONAL,
documentArchitectureClass
[1]      IMPLICIT OCTET STRING OPTIONAL,

```

nonBasicDocCharacteristics	[2]	IMPLICIT NonBasicDocCharacteristics OPTIONAL,
nonBasicStrucCharacteristics	[3]	IMPLICIT NonBasicStrucCharacteristics OPTIONAL }

9 Conformance

For further study.

ANNEX A

(to Recommendation T.433)

Reliable transfer modes

(Informative)

The following figures show the examples of protocol sequence for the reliable transfer modes.

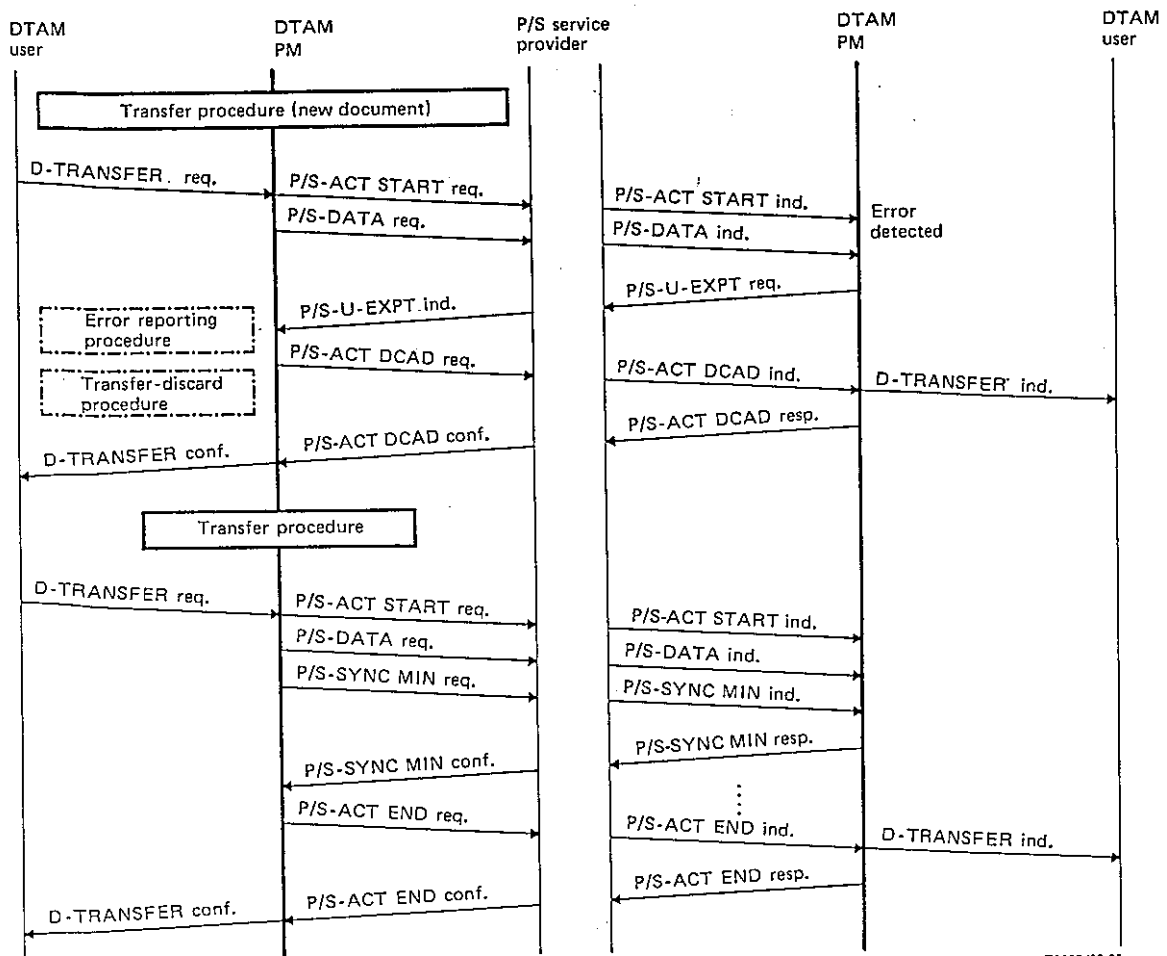


FIGURE A-1/T.433

Transfer procedure (transfer-discard procedure)
(reliable transfer mode 1)

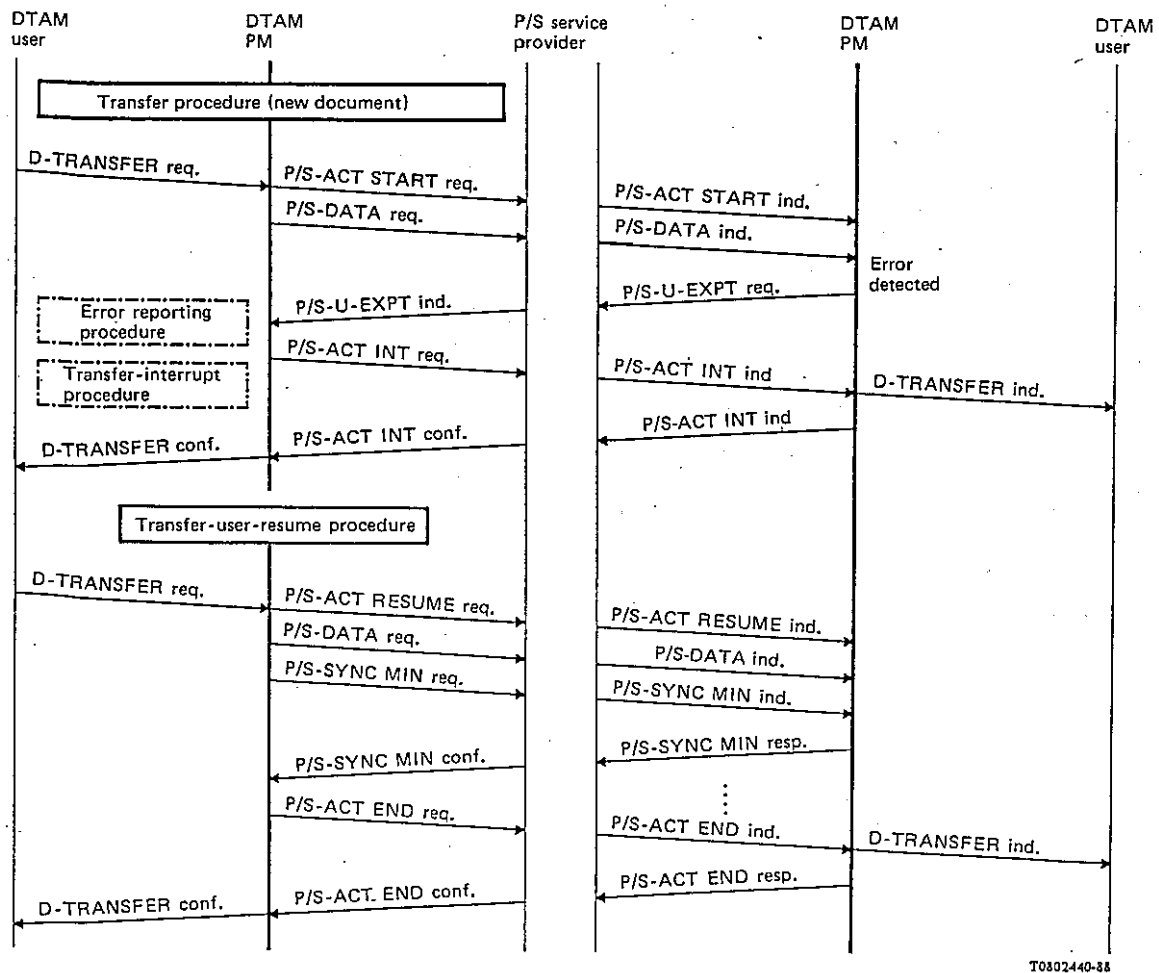


FIGURE A-2/T.433

Transfer procedure (transfer-interrupt procedure) and
transfer-user-resume procedure
(reliable transfer mode 1)

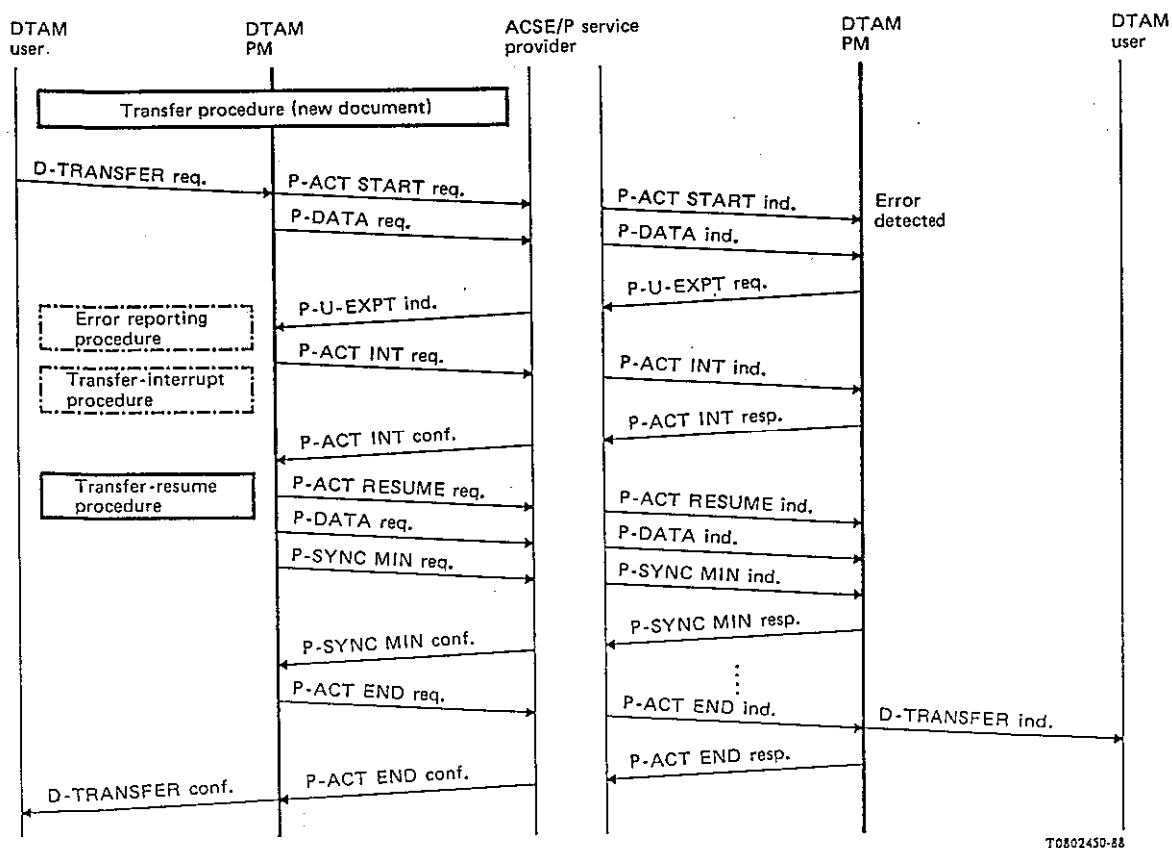


FIGURE A-3/T.433

Transfer procedure (transfer-interrupt procedure) and transfer-resume procedure (reliable transfer mode 2)

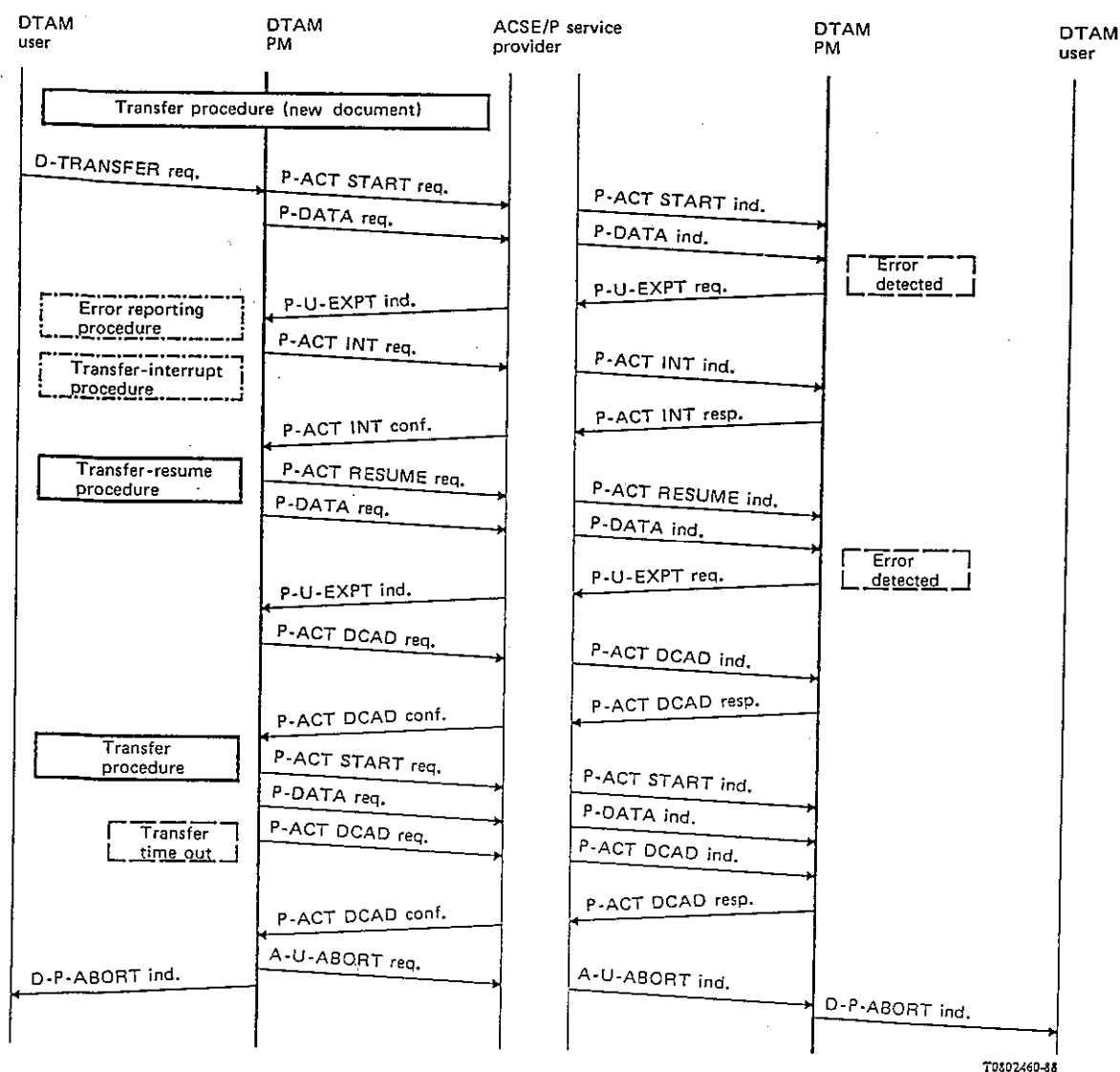


FIGURE A-4/T.433

Transfer procedure, transfer-interrupt procedure,
transfer-resume procedure and transfer time out
(reliable transfer mode 2)

ANNEX B

(to Recommendation T.433)

DTAM-PM state tables

(Transparent mode/reliable transfer mode 1)

This Annex forms an informal part of this Recommendation. This Annex is just for reference. Further study is required.

B.1 *General*

This Annex defines a single DTAM Protocol Machine (DTAM-PM) in terms of a state table. The state table shows the interrelationship between the state of an application-association, the incoming events that occur in the protocol, the actions taken, and finally, the resultant state of the application-association.

The DTAM-PM state table does not constitute a formal definition of a DTAM-PM. It is included to provide a more precise specification of the elements of procedure defined in 6.

This Annex contains the following tables:

- a) Table B-1/T.433 specifies the abbreviated name, source, and name/description of each incoming event. The sources are:
 - 1) DTAM-SE-user (DTAM-SE-user);
 - 2) peer DTAM-PM (DTAM-PM-peer);
 - 3) session service provider (SS-provider);
 - 4) DTAM-PM (DTAM-PM).
- b) Table B-2/T.433 specifies the abbreviated name of each state of the DTAM-PM.
- c) Table B-3/T.433 specifies the abbreviated name, target, and name/description of each outgoing event. The targets are:
 - 1) DTAM-SE-user (DTAM-SE-user);
 - 2) peer DTAM-PM (DTAM-PM-peer);
 - 3) session service provider (SS-provider);
 - 4) DTAM-PM (DTAM-PM).
- d) Table B-4/T.433 specifies the predicates;
- e) Table B-5/T.433 specifies the specific actions;
- f) Table B-6/T.433 through B-14/T.433 including specifies the DTAM-PM state table using the abbreviations of the above tables.

For some events the source and the target is the DTAM-PM (internal event). If the DTAM-PM issues an internal event as part of an action taken, the DTAM-PM awaits that internal event in the resultant state.

TABLE B-1/T.433 (part 1 of 3)

Incoming event list

Abbreviated name	Source	Name and description
D-CAPreq	DTAM-SE-user	D-CAPABILITY request primitive
D-CAPres+	DTAM-SE-user	D-CAPABILITY response primitive (Result = "accepted")
D-CAPres-	DTAM-SE-user	D-CAPABILITY response primitive (Result = "rejected")
D-INTreq	DTAM-SE-user	D-INITIATE request primitive
D-INTres+	DTAM-SE-user	D-INITIATE response primitive (Result = "accepted")
D-INTres-	DTAM-SE-user	D-INITIATE response primitive (Result = "rejected")
D-TERreq	DTAM-SE-user	D-TERMINATE request primitive
D-TERres	DTAM-SE-user	D-TERMINATE response primitive
D-TRreq	DTAM-SE-user	D-TRANSFER request primitive
D-TRreq*	DTAM-SE-user	D-TRANSFER request primitive for resuming purpose
D-TPreq	DTAM-SE-user	D-TOKEN-PLEASE request primitive
D-CGreq	DTAM-SE-user	D-CONTROL GIVE request primitive
D-UAreq	DTAM-SE-user	D-(USER-)ABORT request primitive
DCPQ	DTAM-PM-PEER	D-CAPABILITY-REQUEST APDU as user data of an S-CAPAB-DATA indication primitive
DCPR+	DTAM-PM-PEER	D-CAPABILITY-RESPONSE APDU as user data of an S-CAPAB-DATA confirm primitive (result = "accepted")

TABLE B-1/T.433 (part 1 of 3) (cont.)

Abbreviated name	Source	Name and description
DCPR-	DTAM-PM-PEER	D-CAPABILITY-RESPONSE APDU as user data of an S-CAPAB-DATA confirm primitive (result = "rejected by ...")
DINQ	DTAM-PM-PEER	D-INITIATE-REQUEST APDU as user data of an S-CONNECT indication primitive
DINR+	DTAM-PM-PEER	D-INITIATE-RESPONSE APDU as user data of an S-CONNECT confirm primitive (result = "accepted")
DINR-	DTAM-PM-PEER	D-INITIATE-RESPONSE APDU as user data of an S-CONNECT confirm primitive (result = "rejected by ...")
DAB	DTAM-PM-PEER	D-ABORT APDU as user data of an S-ABORT indication primitive
DTEQ	DTAM-PM-PEER	D-TERMINATE-RESPONSE APDU as user data of an S-RELEASE indication primitive
DTER	DTAM-PM-PEER	D-TERMINATE-RESPONSE APDU as user data of an S-RELEASE confirm primitive
SEG	DTAM-PM-PEER	Segment of document information as user data of an D-DATA indication primitive

TABLE B-1/T.433 (part 2 of 3)

Incoming event list

Abbreviated name	Source	Name and description
S-CONcnf-	SS-provider	S-CONNECT confirm primitive (Result = "rejected") no DINR- APDU
S-RELind	SS-provider	S-RELEASE indication primitive
S-RELcnf	SS-provider	S-RELEASE confirm primitive
S-PABind	SS-provider	S-P-ABORT indication primitive
S-ASind	SS-provider	S-ACTIVITY-START indication primitive
S-MSind	SS-provider	S-MINOR-SYNCHRONIZE indication primitive
S-MScnf	SS-provider	S-MINOR-SYNCHRONIZE confirm primitive
S-AEind	SS-provider	S-ACTIVITY-END indication primitive
S-AEcnf	SS-provider	S-ACTIVITY-END confirm primitive
S-CGind	SS-provider	S-CONTROL-GIVE indication primitive
S-TPind	SS-provider	S-TOKEN-PLEASE indication primitive
S-UEind	SS-provider	S-U-EXCEPTION-REPORT indication primitive
S-PEind	SS-provider	S-P-EXCEPTION-REPORT indication primitive
S-AIind	SS-provider	S-ACTIVITY-INTERRUPT indication primitive
S-AIcnf	SS-provider	S-ACTIVITY-INTERRUPT confirm primitive
S-ADind	SS-provider	S-ACTIVITY-DISCARD indication primitive
S-ADcnf	SS-provider	S-ACTIVITY-DISCARD confirm primitive
S-ARind	SS-provider	S-ACTIVITY-RESUME indication primitive

TABLE B-1/T.433 (part 3 of 3)

Incoming event list

Abbreviated name	Source	Name and description
a-ab	DTAM-PM	association (connection) aborted
a-res	DTAM-PM	activity resumption by the receiving DTAM-PM
a-ret	DTAM-PM	activity completed, discarded, or interrupted
ass-ab	DTAM-PM	start of association (connection)-abort procedure
next	DTAM-PM	transfer of the next segment
p-ab	DTAM-PM	start of provider-abort procedure
tr-discard	DTAM-PM	start of transfer-discard procedure
tr-interr	DTAM-PM	start of transfer-interrupt procedure
tr-p-ab	DTAM-PM	start of procedures transfer-abort followed by provider-abort
tr-pos	DTAM-PM	transfer successful completed
tr-res	DTAM-PM	start of transfer-resumption procedure
transfer	DTAM-PM	start of transfer procedure
resume	DTAM-PM	start of resume procedure
u-exr	DTAM-PM	start of user-exception-report procedure

TABLE B-2/T.433 (part 1 of 2)

DTAM-PM states

Abbreviated name	Name and description
STA0	idle: unassociated
STA01	awaiting DINR+, DINR-, or A-ASCnf-
STA02	awaiting D-INTres+, or D-INTres-
STA11	associated: DTAM-PM is association-initiating DTAM-PM and sending DTAM-PM
STA111	associated: DTAM-PM is association-initiating DTAM-PM and awaiting DCPR+ or DCPR-
STA12	associated: DTAM-PM is association-initiating DTAM-PM and receiving DTAM-PM
STA121	associated: DTAM-PM is association-initiating DTAM-PM and awaiting D-CAPres+ or D-CAPres-
STA21	associated: DTAM-PM is association-responding DTAM-PM and sending DTAM-PM
STA211	associated: DTAM-PM is association-responding DTAM-PM and awaiting DCPR+ or DCPR-
STA22	associated: DTAM-PM is association-responding DTAM-PM and receiving DTAM-PM
STA221	associated: DTAM-PM is association-responding DTAM-PM and awaiting D-CAPres+ or D-CAPres-
STA30	transfer: sending DTAM-PM
STA31	suspended transfer: sending DTAM-PM
STA32	awaiting-S-AEcnf: sending DTAM-PM
STA321*	awaiting tr-pos: sending DTAM-PM
STA34*	awaiting tr-discard to be followed by D-TRcnf+ sending DTAM-PM
STA341	awaiting S-ADcnf to be followed by D-TRcnf+: sending DTAM-PM
STA35*	awaiting tr-discard to be followed by D-TRcnf-: sending DTAM-PM
STA351	awaiting S-ADcnf to be followed by D-TRcnf-: sending DTAM-PM

TABLE B-2/T.433 (part 2 of 2)

DTAM-PM states

Abbreviated name	Name and description
STA37*	awaiting tr-interr to be followed by D-TRcnf-: sending DTAM-PM
STA371	awaiting S-AIcnf: sending DTAM-PM
STA38*	awaiting ass-ab: sending DTAM-PM
STA381*	awaiting a-ab: transfer sending DTAM-PM
STA40	awaiting DTR: transfer receiving DTAM-PM
STA400	awaiting DTR: ignored transfer receiving DTAM-PM
STA41	awaiting S-MSind or S-AEind: transfer receiving DTAM-PM
STA410	awaiting S-MSind or S-AEind: ignored transfer receiving DTAM-PM
STA42	awaiting recovery after u-exr event: transfer receiving DTAM-PM
STA43*	awaiting a-ret: transfer receiving DTAM-PM
STA44*	awaiting u-exr: transfer receiving DTAM-PM
STA45*	awaiting a-res: transfer receiving DTAM-PM
STA48*	awaiting ass-ab: transfer receiving DTAM-PM
STA481*	awaiting a-ab: transfer receiving DTAM-PM
STA70*	awaiting abort: unassociated
STA71*	awaiting abort: associated
STA91	awaiting D-TERres
STA92	awaiting S-RELCnf

TABLE B-3/T.433 (part 1 of 3)

Outgoing event list

Abbreviated name	Target	Name and description
D-CAPind	DTAM-SE-user	D-CAPABILITY indication primitive
D-CAPcnf+	DTAM-SE-user	D-CAPABILITY confirm primitive (Result = "accepted")
D-CAPcnf-	DTAM-SE-user	D-CAPABILITY confirm primitive (Result = "rejected")
D-INTind	DTAM-SE-user	D-INITIATE indication primitive
D-INTcnf+	DTAM-SE-user	D-INITIATE confirm primitive (Result = "accepted")
D-INTcnf-	DTAM-SE-user	D-INITIATE confirm primitive (Result = "rejected")
D-TERind	DTAM-SE-user	D-TERMINATE indication primitive
D-TERcnf	DTAM-SE-user	D-TERMINATE confirm primitive
D-TPind	DTAM-SE-user	D-TOKEN-PLEASE indication primitive
D-TRind	DTAM-SE-user	D-TRANSFER indication primitive
D-TRcnf+	DTAM-SE-user	D-TRANSFER confirm primitive (Result = "Document Information-transferred")
D-TRcnf-	DTAM-SE-user	D-TRANSFER confirm primitive (Result = "Document Information-not-transferred")
D-CGind	DTAM-SE-user	D-CONTROL-GIVE indication primitive
D-UAind	DTAM-SE-user	D-U-ABORT indication primitive
D-PAind	DTAM-SE-user	D-P-ABORT indication primitive

TABLE B-3/T.433 (part 1 of 3) (cont.)

Abbreviated name	Target	Name and description
DCPQ	DTAM-PM-peer	D-CAPABILITY-REQUEST APDU as user data of an S-CAPAB-DATA request primitive
DCPR+	DTAM-PM-peer	D-CAPABILITY-RESPONSE APDU as user data of an S-CAPAB-DATA response primitive (result = "accepted")
DCPR-	DTAM-PM-peer	D-CAPABILITY-RESPONSE APDU as user data of an S-CAPAB-DATA response primitive (result = "rejected by ...")
DINQ	DTAM-PM-peer	D-INITIATE-REQUEST APDU as user data of an S-CONNECT request primitive
DINR+	DTAM-PM-peer	D-INITIATE-RESPONSE APDU as user data of an S-CONNECT response primitive (result = "accepted")
DINR-	DTAM-PM-peer	D-INITIATE-RESPONSE APDU as user data of an S-CONNECT response primitive (result = "rejected by ...")
DAB	DTAM-PM-peer	D-ABORT APDU as user data of an S-ABORT request primitive
DTEQ	DTAM-PM-peer	D-TERMINATE-REQUEST APDU as user data of an S-RELEASE request primitive
DTER	DTAM-PM-peer	D-TERMINATE-RESPONSE APDU as user data of an S-RELEASE response primitive
SEG	DTAM-PM-peer	Segment of Document Information as user data of an D-DATA request primitive

TABLE B-3/T.433 (part 2 of 3)

Outgoing event list

Abbreviated name	Target	Name and description
S-RELreq	SS-provider	S-RELEASE request primitive
S-RELres	SS-provider	S-RELEASE response primitive
S-ASreq	SS-provider	S-ACTIVITY-START request primitive
S-MSreq	SS-provider	S-MINOR-SYNCHRONIZE request primitive
S-MSres	SS-provider	S-MINOR-SYNCHRONIZE response primitive
S-AEreq	SS-provider	S-ACTIVITY-END request primitive
S-AEres	SS-provider	S-ACTIVITY-END response primitive
S-CGreq	SS-provider	S-CONTROL-GIVE request primitive
S-TPreq	SS-provider	S-TOKEN-PLEASE request primitive
S-UEreq	SS-provider	S-U-EXCEPTION-REPORT request primitive
S-AIreq	SS-provider	S-ACTIVITY-INTERRUPT request primitive
S-AIres	SS-provider	S-ACTIVITY-INTERRUPT response primitive
S-ADreq	SS-provider	S-ACTIVITY-DISCARD request primitive
S-ADres	SS-provider	S-ACTIVITY-DISCARD response primitive
S-ARreq	SS-provider	S-ACTIVITY-RESUME request primitive

TABLE B-3/T.433 (part 3 of 3)

Outgoing event list

Abbreviated name	Target	Name and description
a-ab	DTAM-PM	association (connection) aborted
a-res	DTAM-PM	activity resumption by the receiving DTAM-PM
a-ret	DTAM-PM	activity completed, discarded, or interrupted
ass-ab	DTAM-PM	start of association-abort procedure
ass-rec	DTAM-PM	start of association-recovery procedure
next	DTAM-PM	transfer of the next segment
p-ab	DTAM-PM	start of provider-abort procedure
tr-discard	DTAM-PM	start of transfer-discard procedure
tr-interr	DTAM-PM	start transfer-interrupt procedure
tr-p-ab	DTAM-PM	start of procedures transfer-abort followed by provider-abort
tr-pos	DTAM-PM	transfer successful completed
transfer	DTAM-PM	start of transfer procedure
resume	DTAM-PM	start of user-resume procedure
u-exr	DTAM-PM	start of user-exception-report procedure

TABLE B-4/T.433

Predicates

Code	Name and description
p1	DTAM-PM can support the request application-association (connection)
p2	turn assigned to DTAM-PM
p11	association-initiating DTAM-PM
P30	only one segment required to transfer the encoded-APDU-value (no checkpointing)
p31	segment is the last one in a series of segments to transfer the encoded-APDU-value
p32	outstanding-minor-syncs < window-size
p33	outstanding-minor-syncs = 0
p34	sending DTAM-PM is willing to recover from S-PEind
p35	checkpoint-confirmed (at least on S-MScnf received)
p361	reason parameter value of S-UEind is "receiving ability jeopardized"
p362	reason parameter value of S-UEind is "unrecoverable procedure error"
p363	reason parameter value of S-UEind is "non-specific error"
p364	reason parameter value of S-UEind is "sequence error"
p365	reason parameter value of S-UEind is "local SS-user error"
p37	transfer-completed
p41	received segment secured
p42	complete DTAM-SE-user APDU secured
p43	transfer to be resumed was already completed
p44	receiving DTAM-PM is willing to perform and ignore transfer

TABLE B-5/T.433

Specific actions

Code	Name and description
a1	association-initiating DTAM-PM = TRUE
a2	association-initiating DTAM-PM = FALSE
a30	outstanding-minor-syncs = 0, set timer tr to transfer-time, transfer-completed = FALSE, checkpoint-confirmed = FALSE.
a31	outstanding-minor-syncs = outstanding-minor-syncs +1
a32	outstanding-minor-syncs = outstanding-minor-syncs -1 checkpoint-confirmed = TRUE
a41	set reason parameter value of S-UEreq to "sequence error"
a93	transfer-completed = TRUE
a94	transfer-completed = FALSE

TABLE B-6/T.433

DTAM-PM state table: Association-establishment

	STA01	STA01	STA02
D-INTreq	p1: DINQ [a1] STA01		
D-INTres+			DINR+ STA22
D-INTres-			DINR- STA0
DINQ	p1: D-INTind [a2] STA02 ~p1: DINR- STA0		
DINR+		D-INTcnf+ STA11	
DINR-		D-INTcnf- STA0	
D-UAreq		DAB STA0	DAB STA0
DAB		D-UAind STA0	D-UAind STA0
S-PABind		D-PAind STA0	D-PAind STA0

TABLE B-7/T.433 (part 1 of 2)

DTAMPM state table: Association-established, outside transfer

	STA11	STA12	STA21	STA22
D-TRreq	transfer		transfer	
D-TRreq*	resume		resume	
D-CAPreq	DCPQ STA111		DCPQ STA211	
DCPQ		D-CAPind STA121		D-CAPind STA221
D-TERreq	DTEQ STA92			
DTEQ				D-TERind STA91
D-TPreq		S-TPreq STA12		S-TPreq STA22
S-TPind	D-TPind STA11		D-TPind STA21	
D-CGreq	S-CGreq STA12		S-CGreq STA22	
S-CGind		D-CGind STA11		D-CGind STA21
D-UAreq	DAB STA0	DAB STA0	DAB STA0	DAB STA0
DAB	D-UAind STA0	D-UAind STA0	D-UAind STA0	D-UAind STA0
S-PABind	D-PAind STA0	D-PAind STA0	D-PAind STA0	D-PAind STA0

TABLE B-7/T.433 (part 2 of 2)

DTAMPM state table: Association-established, outside transfer

	STA111	STA121	STA211	STA221
D-CAPres+		DCPR+ STA12		DCPR+ STA22
D-CAPres-				
DCPR+	D-CAPcnf+ STA11		D-CAPcnf+ STA21	
DCPR-				
D-UAreq	DAB STA0	DAB STA0	DAB STA0	DAB STA0
DAB	D-UAind STA0	D-UAind STA0	D-UAind STA0	D-UAind STA0
S-PABind	D-PAind STA0	D-PAind STA0	D-PAind STA0	D-PAind STA0

TABLE B-8/T.433 (part 1 of 2)

DTAM-PM, transfer

	STA30	STA31	STA32	STA321*
transfer	p30 [a30] S-ASreq SEG S-AEreq STA32 ~p30: [a30] S-ASreq next STA30			
resume	p30 [a30] S-ARreq SEG S-AEreq STA32 ~p30: [a30] S-ARreq next STA30			
next	p32&~p31: SEG S-MSreq [a31] next STA30 p32&~p31: SEG S-AEreq STA32 ~p32: STA31			
P-MScnf	[a32] STA30	[a32] next STA30	[a32] STA32	
P-AEcnf			p33 tr-pos STA321	
tr-pos				p11: D-TRcnf+ STA11 ~p11: D-TRcnf+ STA21

TABLE B-8/T.433 (part 2 of 2)

DTAM-PM state table: Sending DTAM-PM, transfer

	STA30	STA31	STA32
S-UEind	p361: tr-p-ab STA71 p362: tr-discard STA35 p363: tr-discard STA35 p364: tr-discard STA34 p365&p35: tr-interr STA37 p365&¬p35: tr-discard STA35	p361: tr-p-ab STA71 p362: tr-discard STA35 p363: tr-discard STA35 p364: tr-discard STA34 p365&p35: tr-interr STA37 p365&¬p35: tr-discard STA35	p361: tr-p-ab STA71 p362: tr-discard STA35 p363: tr-discard STA35 p364: tr-discard STA34 p365&p35: tr-interr STA37 p365&¬p35: tr-discard STA35
S-PEind	p34&p35: tr-interr STA37 p34&¬p35: tr-discard STA35 ¬p34: tr-p-ab STA71	p34&p35: tr-interr STA37 p34&¬p35: tr-discard STA35 ¬p34: tr-p-ab STA71	p34&p35: tr-interr STA37 p34&¬p35: tr-discard STA35 ¬p34: tr-p-ab STA71
S-PABind	a-ab STA381	a-ab STA381	a-ab STA381
D-UAreq	DAB STA0	DAB STA0	DAB STA0
DAB	D-UAind STA0	D-UAind STA0	D-UAind STA0

TABLE B-9/T.433

DTAM-PM state table: Sending DTAM-PM, error handling

	STA34*	STA341	STA35*	STA351
tr-discard	S-ADreq STA341		S-ADreq STA351	
D-ADcnf		tr-pos STA321		p11: D-TRcnf- STA11 ¬p11: D-TRcnf- STA21
S-PABind		a-ab STA381		a-ab STA381
D-UAreq		DAB STA0		DAB STA0
DAB		D-UAind STA0		D-UAind STA0

TABLE B-10/T.433

DTAM-PM state table: Sending DTAM-PM error handling

	STA37*	STA371
tr-interr	S-AIreq STA371	
S-AIcnf		p11: D-TRcnf- STA11 ¬p11 D-TRcnf- STA12
S-PABind		a-ab STA381
D-UAreq		DAB STA0
DAB		D-UAind STA0

TABLE B-11/T.433

DTAM-PM state table: Sending DTAM-PM error handling

	STA38*	STA381*
ass-ab	DAB a-ab STA381	
a-ab		D-PAind STA0

TABLE B-12/T.433

DTAM-PM state table: Receiving DTAM-PM

	STA40	STA41	STA400	STA410	STA42
DTNQ	STA41		STA41		
S-MSind		p41: S-MSres STA40		S-MSres STA400	
S-AEind		D-TRind			
S-AIind	S-AIres a-ret STA43	S-AIres a-ret STA43	S-AIres a-ret STA43	S-AIres a-ret STA43	S-AIres a-ret STA43
S-ADind	S-ADres a-ret STA43	S-ADres a-ret STA43	S-ADres a-ret STA43	S-ADres a-ret STA43	S-ADres a-ret STA43
S-PEind	STA40	STA41	STA400	STA410	STA42
S-PABind	a-ab STA481	a-ab STA481	a-ab STA481	a-ab STA481	a-ab STA481
D-TPreq	S-TPreq STA40	S-TPreq STA41	S-TPreq STA400	S-TPreq STA410	
D-UAreq	DAB STA0	DAB STA0	DAB STA0	DAB STA0	DAB STA0
DAB	D-UAind STA0	D-UAind STA0	D-UAind STA0	D-UAind STA0	D-UAind STA0

TABLE B-13/T.433

DTAM-PM state table: Receiving DTAM-PM error handling

	STA43*	STA44*	STA45*	STA48*	STA481*
a-ret	p11: STA12 ¬p11: STA22				
u-exr		S-UEreq [a38] STA42			
a-res			¬p43: STA40 p43&p44: STA400 p43&¬p44: [a41] u-exr STA44		
ass-ab				DAB a-ab STA481	
a-ab					D-PAind STA0

TABLE B-14/T.433

DTAM-PM state table: Abort and association-release

	STA70*	STA71*	STA91	STA92
tr-p-ab	D-TRcnf- D-PAind STA0	D-TRcnf- DAB D-PAind STA0		
p-ab	D-PAind STA0	D-PAind DAB STA0		
D-TERres			S-RELres STA0	
S-RELcnf				D-TERcnf STA0

B.2 Conventions

The intersection of an incoming event (row) and a state (column) forms a cell.

In the state table, a blank cell represents the combination of an incoming event and a state that is not defined for the DTAM-PM (see B.3.1). Some states await solely some incoming events from the source DTAM-PM (internal events). These states are marked by * and no other incoming events are considered.

A non-blank cell represents an incoming event and a state that is defined for the DTAM-PM. Such a cell contains one or more action lists. An action list may be either mandatory or conditional. If a cell contains a mandatory action list, it is the only action list in the cell.

A mandatory action list contains:

- a) optionally one or more outgoing events;
- b) optionally one or more specific actions;
- c) an resultant state.

A conditional action list contains:

- a) a predicate expression comprising predicates and Boolean operators (represents the Boolean NOT, & represents the Boolean AND);
- b) a mandatory action list. (This mandatory action list is used only if the predicate expression is true.)

B.3 Actions to be taken by the DTAM-PM

The DTAM-PM state table defines the action to be taken by the DTAM-PM in terms of an optional outgoing event, optional specific actions, and the resultant state of the application-association.

B.3.1 Invalid intersections

Blank cells indicate an invalid intersection of an incoming event and state. If such an intersection occurs, one of the following actions is taken:

- a) If the incoming event comes from the DTAM-SE-user, or is an internal event, any action taken by the DTAM-PM is a local matter.
- b) If the incoming event is related to a received APDU, or SS-provider, either the DTAM-PM issues an appropriate internal event, or the DTAM-PM issues both a D-PAind outgoing event (to its DTAM-SE-user) and a DAB outgoing event (to its peer DTAM-PM).

B.3.2 Valid intersections

If the intersection of the state and incoming event is valid, one of the following actions is taken:

- a) If the cell contains a mandatory action list, the DTAM-PM takes the actions specified.
- b) If a cell contains one or more conditional action lists, for each predicate expression that is true, the DTAM-PM takes the actions specified. If none of the predicate expressions are true, the DTAM-PM takes one of the actions defined in B.3.1.

B.4 Definition of variables

The following variables are specified.

B.4.1 Association-initiating DTAM-PM

This Boolean variable is set TRUE if the DTAM-PM is the association-initiating DTAM-PM (specific action [a1]), otherwise it is set FALSE (specific action [a2]).

This Boolean variable is tested in the predicate p11.

B.4.2 Checkpoint-confirmed

This Boolean variable is TRUE, if at least one checkpoint was confirmed during the transfer procedure. It is set FALSE at the beginning of the transfer procedure (specific action [a30]). It is set TRUE, if an S-MINOR-SYNCHRONIZE confirm primitive is issued to the sending DTAM-PM (specific action [a32]).

B.4.3 *Transfer-completed*

This Boolean variable is TRUE, if the receiving DTAM-PM aborted the association because it could not discard an already completed transfer. It is set by the specific actions [a93] and [a94].

This Boolean variable is tested in the predicate p37.

B.4.4 *Outstanding-minor-syncs*

This integer variable indicates the number of outstanding checkpoint confirmations during the transfer procedure. It is set to zero at the beginning of the transfer procedure (specific action [a30]). It is incremented by one, if a S-MINOR-SYNCHRONIZE request primitive is issued by the sending user to the sending DTAM-PM (specific action [a31]).

The value of this variable is compared with the value of the window-size field of the S-CONcnf in the predicate p32. The value of this variable is compared with the value zero in the predicate p33.

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure and Internet protocol aspects
Series Z	Languages and general software aspects for telecommunication systems