Network Working Group   ISO
Request for Comments: 892   December 1983

ISO Transport Protocol Specification

This document is distributed as an RFC for information only.
It does not specify a standard for the ARPA Internet.

Note: This document appeared in:

ISO/TC97/SC16/WG6. Information Processing Systems - Open Systems Interconnection - Transport Protocol Specification. Computer Communication Review 12, 3-4 (July/October 1982), pp. 24-67.

and differs from it only in format.

Table of Contents

0. Introduction

The Transport Protocol Standard is one of a set of International Standards produced to facilitate the interconection of computer systems. The set of standards covers the services and protocols required to achieve such interconnection.

The Transport Protocol Standard is positioned with respect to other related standards by the layers defined in the Reference Model for Open Systems Interconnection (ISO 7498). It is most closely related to, and lies within the field of application of the Transport

Service Standard (DP aaaa). It also uses and makes reference to the Network Service Standard (DP bbbb), whose provisions it assumes in order to accomplish the transport protocol's aims. The interrelationship of these standards is depicted in Figure 1.

```
-----------------------------------TRANSPORT SERVICE DEFINITION-----

 Transport                --Reference to aims---------------
 Protocol
 Specification            --Reference to assumptions--------

------------------------------------NETWORK SERVICE DEFINITION------
```

Figure 1 – Relationship between the transport protocol and adjacent services

The standard specifies a common encoding and a number of classes of transport protocol procedures to be used with different network qualities of service.

It is intended that the Transport Protocol should be simple but general enough to cater for the total range of Network Service qualities possible, without restricting future extensions.

The protocol is structured to give rise to classes of protocol which are designed to minimize possible incompatibilities and implementation costs.

The classes are selectable with respect to the Transport and Network Services in providing the required quality of service for the interconnection of two session entities (note that each class provides a different set of functions for enhancement of service qualities).

This protocol standard is concerned with optimisation of network tariffs and the following qualities of service:

a) different throughput rates;
b) different error rates;
c) integrity of data requirements;
d) reliability requirements.

The aim of this standard is primarily to provide a definition for implementors. Since the protocol is complex, the document contains much material which is advisory or descriptive, but mandatory requirements on implementations are clearly identified.

It should be noted that, as the number of valid protocol sequences is very large, it is not possible with current technology to verify that an implementation will operate the protocol defined in this document correctly under all circumstances. It is possible by means of testing to establish confidence that an implementation correctly operates the protocol in a representative sample of circumstances. It is, however, intended that this standard can be used in circumstances where two implementations fail to communicate in order to determine whether one or both have failed to operate the protocol correctly.

The variations and options available within this standard are essential to enable a Transport Service to be provided for a wide variety of applications over a variety of network qualities. Thus, a minimally conforming implementation will not be suitable for use in all possible circumstances. It is important therefore to qualify all references to this standard with statements of the options provided or required or with statements of the intended purpose of provision or use.

1. Scope and Field of Application

1.1 This International Standard Specifies:

a) five classes of procedures

1) Class 0. Simple class;
2) Class 1. Basic error recovery class;
3) Class 2. Multiplexing class;
4) Class 3. Error recovery class;
5) Class 4. Error detection and recovery class,

for the transfer of data and control information from one transport entity to a peer transport entity;

b) the means of negotiating the class of procedures to be used by the transport entities;

c) the encoding of the transport protocol data units used for the transfer of data and control information;

d) the functional requirements of equipment within a computer system claiming to implement these procedures.

1.2 The procedures are defined in terms of:

a) the interactions between peer transport entities through the exchange of transport protocol data units;

b) the interactions between a transport entity and the transport service user in the same system through the exchange o
f
transport service primitives;

c) the interactions between a transport entity and the network service provider through the exchange of network service primitives.

1.3 This International Standard is applicable to equipment which supports the Transport Layer of the OSI Reference Model and which wishes to interconnect in an open systems environment.

2. References

ISO 7498 Information processing systems - Open systems interconnection - Basic Reference Model

DP aaaa Information processing systems - Open systems interconnection - Transport service definition (N1169).

DP bbbb Information processing systems - Open systems interconnection - Connection-oriented network service definition (N729)

Section One - General

3. Definitions

3.1 Equipment: Hardware or software or a combination of both; it need not be physically distinct within a computer system.

3.2 Transport service user: An abstract representation of the totality of those entities within a single system that make use of the transport service.

3.3 Network service provider: An abstract machine which models the totality of the entities providing the network service, as viewed by a transport entity.

Explanatory Notes

1. Definitions 3.1 to 3.3 relate to terms used in clause 1.

2. This standard makes use of the terms, concepts, and definition defined in ISO 7498.

4. Symbols and Abbreviations

4.1 Data Units

TPDU	Transport protocol data unit
TSDU	Transport service data unit
NSDU	Network service data unit

4.2 Types of transport protocol data units

CR TPDU	Connection request TPDU
CC TPDU	Connection confirm TPDU
DR TPDU	Disconnect request TPDU
DC TPDU	Disconnect confirm TPDU
DT TPDU	Data TPDU
ED TPDU	Expedited data TPDU
AK TPDU	Data acknowledge TPDU
EA TPDU	Expedited acknowledge TPDU
RJ TPDU	Rejected TPDU
ERR TPDU	Error TPDU

4.3 TPDU fields

LI	Length indicator (field)
CDT	Credit (field)
TSAP-ID	Transport service access point identifier (field)
DST-REF	Destination reference (field)
SCE-REF	Source reference (field)
EOT	End of TSDU mark
TPDU-NR	DT TPDU number (field)
ED-TPDU-NR	ED TPDU number (field)
YR-TU-NR	Sequence number response (field)

4.4 Parameters

T (R)	Receive sequence number
T (S)	Send sequence number

4.5 Timer variables

T1	Elapse time between retransmissions
N	The maximum number of retransmissions
L	Bound for the maximum time between the decision to transmit a TPDU and the receipt of any response relating to it
T-WAIT	Maximum time for a reassignment to take place before TC failure is assumed
I	Inactivity timer - Maximum time allowed to elapse between receipt of TPDUs before TC failure is assumed
W	Window timer - Maximum interval between transmission of up to date window information

4.6 Other variables

n	The number of bits in the sequence number field
p	The number of bits in the credit field of a CR, CC or AK TPDU

4.7 Miscellaneous

TS-user	Transport service user
TSAP	Transport service access point
NSAP	Network service access point
TC	Transport connection
NC	Network Connection

5. Overview of the Transport Protocol

5.1 Service Provided by the Transport Layer

The services provided by the protocol described in this document are connection-oriented services. They are defined in document DP aaaa. The Transport Service primitives provided are summarized in Figure 1.

Primitive		Parameters
T-CONNECT	Request Indication	To Transport Address, From Transport Address, Expedited Data Option, Quality of Service, TS-User data.
T-CONNECT	Response Confirmation	Responding Address, Quality of Service, Expedited Data Option, TS-User data.
T-DATA	Request Indication	TS-User data.
T-EXPEDITED DATA	Request Indication	TS-User data.
T-DISCONNECT	Request	TS-User data.
T-DISCONNECT	Indication	Disconnect reason, TS-User data.

Figure 1. Transport Service Primitives

5.2 Service Assumed from the Network Layer

The transport protocol described in this document assumes of the Network Services described in DP bbbb. The Network Service primitives used are summarized in Figure 2.

Primitive		X/Y	Parameters	X/Y/Z
N-CONNECT	Request	X	Called Address,	X
	Indication	X	Calling Address,	X
	Response	X	NS-User data,	Z
	Confirmation	X	QOS.	X
N-DATA	Request	X	NS-User data,	X
	Indication	X	Conf. Request	Y
N-DATA ACKNOWLEDGE	Request	Y		
	Indication			
N-EXPEDITED DATA	Request	Y		
	Indication		NS-User data	Y
N-RESET	Request	X		
	Indication	X		
	Response	X		
	Confirmation	X		
N-DISCONNECT	Request	X	NS-User data	Z
	Indication	X		

X - The Transport Protocol assumes that this facility is provided in all networks.

Y - The Transport Protocol assumes that this facility is provided in some networks and a mechanism is provided to optionally use the facility.

Z - The Transport Protocol does not use this parameter.

Figure 2. Network Service Primitives

5.3 Functions of the Transport Layer

5.3.1 Connection Oriented Functions

5.3.1.1 Overview of Functions

The functions in the transport layer are at least those necessary to bridge the gap between the services available from the network layer and those to be offered to the transport users.

The functions in the transport layer are concerned with the enhancement of quality of service, including all aspects of cost optimization. They are described below; the descriptions are grouped into those concerned with the establishment phase, the data transfer phase, and the release phase.

5.3.1.1.1 Establishment Phase

The goal of the establishment phase is to establish a transport connection, i.e., between two transport users. The functions of transport layer during this phase must match the requested class of services with the services provided by the network layer as follows:

- o Select network service which best matches the requirement of the TS-user taking into account charges for various services.

- o Decide whether to multiplex multiple transport connection onto a single network connection.

- o Establish the optimum TPDU size.

- o Select the functions that will be operational upon entering the data transfer phase.

- o Map transport addresses onto network addresses.

- o Provide a means to distinguish between two different transport connections.

- o Transportation of user's data.

5.3.1.1.2 Data Transfer Phase

The purpose of the data transfer phase is to permit two-way simultaneous transport of TSDUs between the session entities connected by the transport connection. This purpose is achieved by means of two-way simultaneous communication in the Transport protocol and by the following functions. Each of these functions is used or not used in accordance with the result of the selection performed in the establishment phase.

- o Concatenation and Separation

  A function used to collect several TPDUs into a single NSDU; the destination transport entity separates the TPDUs.

- o Segmenting and Reassembling

  The splitting of a single data TSDU into multiple TPDUs which are reassembled into their original format at the destination.

- o Multiplexing and Demultiplexing

  A function used to share a single network connection between two or more transport connections.

- o Splitting and Recombing

  A function allowing the simultaneous use of two or more network connections to support the same transport connection.

- o Flow Control

  A function used to regulate the flow of TPDUs between two transport entities on one transport connection.

- o Error Detection

  A function used to detect the loss, corruption, duplication, misordering or misdelivery of TPDUs.

- o Transport Connection Identification

  A means to uniquely identify a transport connection between the pair of transport entities supporting the connection during the lifetime of the transport connection.

- o Error Recovery

  A function used to recover from detected and signalled errors.

- o Expedited Data

  A function used to bypass the flow control of normal data TPDU. Expedited data TPDUs’ flow is controlled by separate flow control.

- o TSDU Delimiting

  A function used to determine the beginning and ending of a TSDU.

5.3.1.1.3 Release Phase

A function to provide a disconnection of the transport connection, regardless of the current activity.

5.3.1.2 Classes and Options

A class defines a set of functions. In this protocol five classes are defined:

- o Class 0: Simple Class
- o Class 1: Basic Error Recovery Class
- o Class 2: Multiplexing Class
- o Class 3: Error Recovery and Multiplexing Class
- o Class 4: Error Detection and Recovery Class.

Note that with the exception of classes 0 and 1, transport connections of different class may be multiplexed together onto the same network connection.

5.3.1.2.2 Options within Classes

Options define potential functions which may be used within a class.

5.3.1.2.3 Negotiation

Classes and options within classes are negotiated during the connection establishment phase.

5.3.1.2.4 Choice of the Class of Protocol

The choice will be made by the transport entities according to:

- o the users requirement expressed via T-CONNECT service primitives. In particular, for the choice of the class of protocol, the following rules apply:

  - if the TS-User requests either transmission of user data during the connection phase, or use of Expedited data transfer, then Class 0 cannot be selected.

  - if the TS-User requests use of Expedited data transfer, then Class 2 with the non-explicit flow control option cannot be selected.

- o the quality of the available Network services;

- o the user required service versus cost ratio acceptable for the transport user.

The following is a classification of network services in terms of quality with respect to error behavior relative to the user requirements. Its main purpose is to provide a basis for the decision regarding which class of transport connection should be used on top of a given network connection.

Type A: Network connection with acceptable residual error rate (for example not signalled by 'clear' or 'reset') and acceptable rate of signalled failures.

Type B: Network connections with acceptable residual error rate (for example not signalled by 'clear' or 'reset') but unacceptable rate of signalled failures.

Type C: Network connections with residual error rate not acceptable to the TS-user.

It is assumed that each transport entity is aware of the quality of service provided by particular Network connections.

5.3.1.3 Potential Functions

The protocol described in this document does not include the following set of functions which have been identified as potential transport layer functions:

- o provision for encryption
- o provision for accounting mechanisms
- o provision for status exchanges and monitoring of quality of service
- o provision for blocking
- o temporary release of network connections

5.4 Model of the Transport Layer

```
     TSAP                              TSAP

Transport Protocol              Transport Protocol
     Entity                          Entity


      NSAP   -------                 NSAP  -------
       |      (NSAP)                  |      (NSAP)
       |       |                      |        |
       |       |-------------------------|---------
       |                              |
      -----------------------------------
```

A Transport Protocol entity within the Transport Layer communicates with a Transport User through a TSAP by means of the service primitives as defined by the transport service definition DP aaaa. Service primitives will cause or be the result of Transport Protocol Data Unit exchanges between the peer Transport Protocol entities supporting a Transport Connection. These protocol exchanges are effected using the services of the Network Layer as defined by the Network Service Definition DP bbbb through one or more NSAPs.

Transport connection endpoints are identified in end systems by an internal, implementation dependent, mechanism so that the Transport User and the Transport Protocol entity can refer to each Transport connection.

Section Two - Transport Protocol Specification

6. Protocol Mechanisms

Several functions are described as 'inherent' or 'pervasive'. Inherent functions must be invoked for every transport connection. Pervasive functions are optional, but if one is invoked for the first transport connection over a network connection, it must also be invoked for any and all other transport connections which use that network connection during its lifetime.

6.1 Assignment to Network Connection

Purpose: Assignment of transport connections to network connections.

Network Service Primitives:

N-CONNECT
N-DISCONNECT

Description:

This function is inherent.

Before a transport connection can be created or used, it must be assigned to one (or more if splitting function is being used) network connection(s). Both transport entities involved must become aware of this assignment. A transport connection may be assigned to a suitable existing network connection; one or more new network connections may also be created for the purpose.

An existing network connection, which connects the relevant transport entities, is unsuitable for assignment of a transport connection if, for example:

- o the quality of service needed for the transport connection can not be met by using or enhancing the network connection.

- o the protocol class preferred or in use for the transport connection is incompatible with the current usage of the network connection as regards the use of pervasive functions (e.g., multiplexing).

When a new network connection is created, the quality of service requested is a local matter, though it will normally be related to the requirements of transport connection(s) expected to be assigned to it.

A Network Connection with no transport connections will be available after initial establishment or because explicit disconnection of all the transport connections previously assigned to it has taken place. Either Transport entity may as a local matter choose to disconnect the Network Connection or assign other Transport Connections to it.

6.2 Transport Protocol Data Unit (TPDU) Transfer

Purpose: To convey transport protocol data unit in user data fields of network service primitives.

Network Service Primitives

N-DATA
N-EXPEDITED DATA

Description:

This function is inherent.

The Transport Protocol Data Units (TPDUs) defined for the protocol are listed in Figure 3.

TPDU name	Abbreviation
Connection Request	CR
Connection Confirm	CC
Disconnect Request	DR
Disconnect Confirm	DC
Data	DT
Expedited Data	ED
Data Acknowledge	AK
Expedited Acknowledge	EA
Reject	RJ
TPDU Error	ERR

Figure 3. Transport Protocol Data Units

TPDUs are conveyed using the NS-User data parameters of the Network Service primitives, primarily with the N-DATA, but also with N-EXPEDITED primitives.

Transport entities shall accept all permissible assignments and may issue any permissible assignments. The permissible assignments of TPDUs to these primitives are shown in Figure 4. Concatenation of TPDUs is also permitted (see section 6.4).

Primitive	Applicable TPDUs	Note
N-DATA	CR, CC, DR, DT, ED, AK, EA, RJ, DC, ERR	
N-EXPEDITED	ED, EA	1

Notes:

1. This assignment is permissible only when using class 1 and when the network expedited variant has been agreed.

Figure 4. Network Service Primitives which can convey TPDUs.

6.3 Data TPDU Length and Segmenting

Purpose: Mapping between one TSDU and TPDUs.

TPDUs and fields used:

DT
- End of TSDU (1 bit)

Description:

The data field of Data TPDUs may contain any number of octets up to an agreed maximum as negotiated at connection time.

A transport entity uses an End of TSDU mark as defined below:

In each Data TPDU a transport entity may indicate the end of a TSDU.

Category 1 Having the End of TSDU mark set to yes. These TPDUs may or may not have the maximum length.

Category 2 Having the End of TSDU mark set to no. These TPDUs do not necessarily have the maximum length.

A complete Data TPDU sequence is defined as being composed of either a single category 1 DT TPDU or consecutive category 2 followed by a category 1 DT TPDU.

6.4 Concatenation and Separation

Pupose: Conveyance of multiple TPDUs in one NSDU.

Description:

All TPDUs carry in their TPDU header a length indicator (see Section 8.2.1). Additionally, TPDUs are classified as either Data TPDUs or Control TPDUs. Control TPDUs may or may not contain a data field. For TPDUs containing data the length of the data field is indicated by the length of the NSDU. These provisions permit any number of Control TPDUs that may not contain data to be concatenated with a single control TPDU which may contain data or with a single Data TPDU. The control TPDUs without data must precede the TPDU with data, if any. The number of TPDUs so concatenated is terminated by the end of the NSDU.

The concatenated set of TPDUs may be for the same or different transport connections. An implementation shall accept concatenated TPDUs and may concatenate TPDUs before transmission. The transport entity shall not send a concatenated set of TPDUs which exceeds twice the overall maximum TPDU length for all the TCs assigned to the network connection.

6.5 Connection Establishment

Purpose: Creation of a new transport connection.

Network Service Primitives:

N-DATA

TPDUs and fields used:

CR, CC
- source reference (16 bits)
- initial credit (if applicable)
- calling transport address (optional)
- called transport address (optional)
- user data (optional)
- TPDU size (optional)
- sequence number length (optional)
- checksum selection (optional)
- acknowledgement time (optional)
- quality of service (optional)

CR
- preferred protocol class

- alternative protocol classes (zero or more)
- version number (optional)
- security (optional)
- proposed options

CC

- destination reference (16 bits)
- selected protocol class
- selected options

Description:

This function is inherent:

A transport connection is established by means of one transport entity (the initiator) transmitting a Connection Request (CR) TPDU to the other transport entity (the responder), which replies with a Connection Confirm (CC) TPDU. Before sending the CR TPDU, the initiator assigns the transport connection being created to one (or more if the splitting function is being used) network connection(s). It is this set of network connections over which the TPDUs are sent. During this exchange, all information and parameters needed for the transport entities to operate must be exchanged or negotiated.

The following information is exchanged:

o references. Each transport entity chooses a reference which is 16 bits long and which is arbitrary except for the following restrictions:

  - it cannot already be in use or "frozen" (see "Frozen References", Section 6.19).

  - it cannot be zero.

Each transport entity is responsible for selecting the Reference which the partner will use. This mechanism is symmetrical and therefore avoids the need to assign a status of master or slave to partners and avoids call collision. This mechanism also provides identification of the transport connection independent of the network connection. The range of References used for transport connections, in a given transport entity, is a local system parameter.

o addresses (optional). Indicate the calling and called transport service access points. When either network address unambiguously defines the transport address this information may be omitted.

o initial credit. Only relevant for classes which include the Explicit Flow Control Function.

o user data. Not available in class 0. Up to 32 octets in in other classes.

The following negotiations take place:

o protocol class. The initiator shall propose a preferred class and any number of alternatives. (Except that no alternatives are allowed when class 0 is the preference.) The initiator should assume when it sends the CR TPDU that its preferred class will be agreed to, and commence the functions associated with that class.

Note: This means, for example, that when a class which includes resynchronization (see "Resynchronization", Section 6.15) is preferred, resynchronization will occur if a reset is signalled during connection establishment.

When the responder has decided which class is to be used, it shall indicate this in the CC TPDU and shall invoke the appropriate functions for the class. The responder may select the preferred class, or any of the alternative classes or may select class 0 if class 1 is proposed or class 2 if class 3 or 4 is proposed. (see Section 9)

If the preferred class is not selected, then on receipt of the CC TPDU, the initiator shall adjust its functions accordingly.

o TPDU Size. The initiator may propose a maximum size for TPDUs, and the responder may accept this value or respond with any value between the proposed value and 128 in the set of values available (see "Encoding", Section 8).

o sequence number length. Either normal or extended is available. When the sequence number is extended, the credit field (if applicable) is also extended.

o checksum selection. This defines whether or not TPDUs of the connection are to include a checksum.

o version number. This defines the version of the transport protocol standard used for this connection.

o security parameter. This parameter and its semantics are user defined.

o quality of service parameter. This defines the throughput, delay, priority and residual error rate.

o The non-use of explicit flow control in class 2 is negotiated.

o The use of Network Receipt Confirmation and Network expedited is negotiated when class 1 is to be used.

The negotiation rules for the options are such that the initiator may propose either to use or not to use the option. The responder may either accept the proposed choice or select the mandatory alternative defined in Section 9.

During the establishment phase of the transport connection, the use of the expedited data option field of CR/CC allows both Transport Service user to negotiate the use or non use of the expedited data transport service as described in the transport service definitions.

The following table summarizes the negotiation possibilities for the options.

Option	Proposition Made by the Initiator	Possible Selection by the Responder
Transport expedited data transfer service	Yes	Yes or No
	No	No
Use of receipt confirmation (class 1 only)	Yes	Yes or No
	No	No
Use of the network expedited variant (class 1 only)	Yes	Yes or No
	No	No
Non use of checksum (class 4 only)	Yes	Yes or No
	No	No
Non use of explicit flow control (class 2 only)	Yes	Yes or No
	No	No
Use of extended format	Yes	Yes or No
	No	No

In class 2, whenever a transport entity requests or agrees to the Transport Expedited data transfer service or to the use of extended formats, it must also request or agree (respectively) to the use of explicit flow control.

6.6 Connection Refusal

Purpose: Refusal of the transport connection.

TPDUs and fields used:

DR
- reason (1 octet)
- user data (maximum of 64 octets)

ERR
- reject code (1 octet)
- rejected TPDU parameter

Description:

If a transport connection cannot be accepted, the called transport entity shall respond to the CR TPDU with a DR TPDU. The clearing reason shall indicate why the connection was not accepted. The source reference field in the DR TPDU is set to zero to indicate an unassigned reference.

If the CR is regarded as an invalid TPDU, the called transport entity will respond by sending an ERR TPDU. On receipt of this TPDU, the calling entity will regard the connection as closed.

6.7 Release

Variants: 'implicit' or 'explicit'

Purpose: Termination of the transport connection.

Network Service Primitives:

N-DISCONNECT (implicit variant only)
N-DATA

TPDUs and fields used:

DR
- clearing reason (1 octet)
- user data (maximum of 64 octets)

DC

Description:

This function is inherent.

In the 'implicit' variant, either transport entity disconnects a transport connection by disconnecting the network connection to which it is assigned. Similarly when a transport entity is informed that the network connection has been disconnected by the peer transport entity, this should be considered as a transport disconnect.

In the 'explicit' variant, either transport entity transmits a Disconnect Request (DR) TPDU, and the other responds with a Disconnect Confirm (DC) TPDU. When the DC TPDU is sent or received by a transport entity, that entity should consider the transport connection not to exist (note 1). After the sending of a DR TPDU, other TPDUs received before the DC TPDU are ignored. It is possible that a disconnect collision will occur, when both transport entities send a DR TPDU at about the same time. This results in each transport entity receiving a DR, after sending one. Each transport entity shall consider the received DR TPDU as a confirmation of its DR TPDU, and shall not send or expect to receive a DC TPDU.

The DR can convey a limited amount (up to 64 octets) of data.

6.8 Implicit Termination

Purpose: Termination of a Transport Connection on the occurrence of a signalled error for which recovery functions are not operative.

Network Service Primitives:

N-DISCONNECT Indication
N-RESET Indication

Description:

When, on the network connection to which a Transport Connection is assigned, an N-DISCONNECT or N-RESET Indication occurs, both transport entities shall consider that the transport connection no longer exists, and so inform the session entities.

Note 1:

When a connection has been released, after the exchange of DR and DC, the reference can be re-used immediately (except in Class 4, where the Frozen Reference function is used, see Section 6.19). This is because the releasing transport entity does not know with certainty that the remote transport entity considers use of the reference to be ended. Therefore, the reference should not be re-used for further connections. (In practice, the reference may be re-used after a reasonable period when it is possible to be reasonably certain that the remote transport entity will not continue to use it).

6.9 Spurious Disconnect

Purpose: To deal with the arrival of an "unknown" DR TPDU.

TPDUs and fields used:

DR, DC
- source reference
- destination reference

Description:

A DR TPDU can be received for a transport connection which does not exist. Rather than treating this as an error, a DC TPDU should be send back which reflects the references of the DR TPDU.

Note:
This only applies when one or more transport connections using a multiplexing class exist over the network connection, or when no transport connections exist. At other times it is a protocol error.

6.10 Data TPDU Numbering

Variants: 'normal' or 'extended'

Purpose: Numbering of DT TPDUs for use in recovery, flow control, or sequencing functions.

TPDUs and Fields Used:

DT
- TPDU-NR (7 or 31 bits)

Description:

DT TPDUs transmitted in each direction on a transport connection bear a sequence number 'TPDU-NR'. Its value in the first DT TPDU in each direction after connection establishment will be zero. Thereafter each TPDU had 'TPDU-NR' one greater than the previous. Modulo 2**7 arithmetic is used in the 'normal' variant, and modulo 2**31 in the 'extended' variant.

In the sections that follow, the relationships 'greater than' and 'less than' are used in connection with TPDU numbers. In all such uses, the numbers being compared cover a range less than the modulus and in fact lie within a contiguous set of TPDU numbers called a 'window'. The window has a known starting TPDU number and finishing number. The term 'less than' means 'occurring sooner in the window sequence' and the term 'greater than' means 'occurring later in the window sequence'.

6.11 Expedited Data Transfer

Variants: 'network expedited' or not

Purpose: Provision of the expedited data service

Network Service Primitives:

N-DATA
N-EXPEDITED DATA

TPDUs and Fields Used:

ED
- ED TPDU-NR (7 or 31 bits)

EA
- YR-TU-NR (7 or 31 bits)

Description:

Each expedited TSDU is conveyed as the data field of an Expedited Data (ED) TPDU.

Each ED TPDU received must be acknowledged by an Expedited Acknowledge (EA) TPDU.

There may only be one ED TPDU unacknowledged at any time for each direction of a transport connection.

In the 'network expedited' variant (available in class 1 only), ED and EA TPDUs are conveyed in the data fields of N-EXPEDITED DATA primitives. Otherwise, N-DATA is used.

6.12 Reassignment

Purpose: Assignment of a Transport Connection to a different Network Connection.

TPDUs and Fields Used:

CR
- source reference

RJ, DR
- destination reference

Description:

When the Network Connection to which a Transport Connection was assigned no longer exists, the Transport Connection can be assigned to another Network Connection.

When one transport entity has assigned the Transport Connection, it is important that the other transport entity recognise to which Network Connection it has been assigned. This can only take place when it has received a TPDU for the Transport Connection on a Network Connection with calling and called network addresses which imply the same transport entities as the old. The TPDU will have been sent as a result of the assigning transport entity commencing resynchronization, and will thus be a RJ, or a retransmitted CR or DR.

The Transport Connection shall be recognised as having been assigned to the Network Connection on which the TPDU was received.

6.13 Reassignment After Failure

Purpose: Recovery from network provider initiated disconnect.

Network Service Primitives:

N-DISCONNECT Indication

Description:

When a N-DISCONNECT Indication arrives for the network connection to which a transport connection is assigned, the transport connection must be reassigned by its initiator (see "Reassignment")

If the reassignment has not successfully occurred within a time of T-wait seconds, then the transport connection must be considered as non-existent by both transport entities.1

1. The CR TPDU does not have a destination reference; nevertheless it can be distinguished from a new connection attempt by having the same source reference.

NOTE: The value of T-wait has to be agreed by the communicating transport entities.

6.14 Retention Until Acknowledgement of TPDUs

Variants: 'confirmation of receipt' or 'AK'

Purpose: To enable and minimize retransmission after possible loss of TPDUs.

Network Service Primitives:

N-DATA
N-DATA ACKNOWLEDGE

TPDUs and Fields Used:

CR, CC, DR, DC

RJ, AK, EA
- YR-TU-NR (7 or 31 bits)

DT
- TPDU-NR (7 or 31 bits)

ED
- ED TPDU-NR (7 or 31 bits)

Description:

Copies of the following TPDUs shall be retained upon transmission to permit their later retransmission:

CR, CC, DR, DT, ED.

NOTE: If DR is sent in response to CR there is no need to retain a copy of the DR.

In the 'confirmation of receipt' variant, applicable only in Class 1, transport entities receiving N-DATA Indications which convey DT TPDUs and have the confirmation request field set shall issue a N-DATA Acknowledge Request at the earliest possible opportunity (1).

(1) It is a local matter for each transport entity to decide which N-DATA Requests should have the confirmation request parameter set. This decision will normally be related to the amount of storage available for retained copies of the DT TPDUs. Use of the confirmation request parameter may affect the quality of network service.

After each TPDU is acknowledged, as shown in Figure 5, the copy need not be retained. Copies may also be discarded when the transport connection ceases to exist.

TPDU	ACKNOWLEDGED BY
CR	receipt of CC, DR, or ERR, TPDU
DR	receipt of DC or DR (in case of collision) TPDU
CC	receipt of RJ, DT, AK, ED, EA TPDUs (or N-DATA ACKNOWLEDGE Indication.)
DT (Note 1)	N-DATA ACKNOWLEDGE Indication when the DT TPDU was sent before or with the oldest N-DATA which had the confirmation request

field set.

DT (Note 2)	receipt of Data Acknowledge (AK) or Reject (RJ) TPDU for which 'YR-TU-NR' is greater than 'TPDU-NR' in the DT TPDU.
ED	receipt of EA TPDU for which 'YR-TU-NR' is equal to 'ED-TPDU-NR' in the ED TPDU.

Notes:

1. Applies to 'confirmation of receipt' variant.
2. Applies to 'AK' variant.

Figure 5. Acknowledgement of TPDUs

6.15 Resynchronization

Purpose: To restore the connection to normal after an error.

Network Service Primitives:

N-RESET Indication

TPDUs and Fields Used:

CR, DR, CC, DC

RJ, EA
- YR-TU-NR (7 or 31 bits)

DT
- TPDU-NR (7 or 31 bits)

ED
- ED TPDU-NR (7 or 31 bits)

Description:

After the reset of an underlying network connection, the resynchronization procedures below are carried out by both transport entities.

After a network connection failure, the reassignment after failure function is invoked and then the resynchronization function. The sequence of events at the two transport entities is the following:

Events at the transport entity initiating reassignment: (the transport entity immediately commences resynchronization by sending a TPDU)

- o if a CR is retained then retransmit it.

- o if a DR is retained then retransmit it.

- o otherwise, resynchronize data:

  - send RJ TPDU with 'YR-TU-NR' field set to the 'TPDU-NR' of the first unreceived DT TPDU

  - when RJ TPDU has been received retransmit any ED TPDUs then DT TPDUs which are unacknowledged

  - any ED TPDUs received which are duplicates shall be acknowledged (by EA TPDUs) and discarded.

Events at the other transport entity:

The transport entity shall not send any TPDUs until after receipt of the TPDU which commenced resynchronization. This TPDU therefore serves two purposes, namely indication of re-assignment and commencement of resynchronization.

- o if the first received TPDU os a DR, then transmit a DC TPDU.

- o if the first received TPDU is a CR and the transport connection is not idle, this means that a CC TPDU is retained: then retransmit it followed by any ED TPDU and then DT TPDUs which are outstanding (that may or may not have been transmitted previously).

NOTE: no TPDUs can be transmitted using network expedited until CC becomes acknowledged, to prevent the network expedited overtaking the CC.

- o if the first received TPDU is a RJ, then act as follows:

  - if a DR TPDU is retained, then retransmit it

  - if a CC TPDU remains unacknowledged, then carry out the data resynchronization procedure described below

  - otherwise resynchronize data:

    - send RJ TPDU with 'YR-TU-NR' field set to the 'TPDU-NR' of the first unreceived DT TPDU

- retransmit any ED TPDUs then DT TPDUs which are unacknowledged
- any ED TPDUs received which are duplicates should be acknowledged (by EA TPDUs) and discarded.

NOTE: It is possible for a transport entity using the Class 1 protocol to decide on a local basis to issue an N-RESET Request. The effect of this request at the remote transport entity is to force it to perform the resynchronization mechanism. This possibility may be used to remove congestion within the network connection.

6.16 Multiplexing and Demultiplexing

Purpose: Concurrent sharing of a network connection by several transport connections.

TPDUs and Fields Used:

CC, DR, DC, DT, AK, ED, EA, RJ, ERR
- destination reference

Description:

This function is pervasive.

When this function is in operation, more than one transport connection can be simultaneously assigned to the same network connection.

Every TPDU (including DT TPDUs) must carry the destination reference, to identify the transport connection to which it refers.

6.17 Explicit Flow Control

Purpose: Regulation of flow of DT TPDUs independently of the flow control in the other layers.

TPDUs and Fields Used:

CR, CC, AK, RJ
- CDT (4 or 16 bits)

DT
- TPDU-NR (7 or 31 bits)

AK, RJ
- YR-TU-NR (7 or 31 bits)
- subsequence number (optional)
- flow control confirmation (optional)

Description:

The mechanism depends on the class. Thus the description can be found in the section describing the class.

6.18 Checksum

Purpose: To detect corruption of TPDUs by the network service provider.

TPDUs and Fields Used:

All TPDUs
- checksum (16 bits - 32 bits)

Description:

When a TPDU is to be transmited for a TC which has selected the checksum option, the sending transport entity must generate a checksum for the TPDU and store it in the checksum parameter in the variable part of the TPDU header. The checksum must be generated as follows:

1. Set up the complete TPDU, including the header and user data (if any). The header must include the checksum parameter in its variable part. The value field of the checksum parameter must be set to zero at this point.

2. Initialize two variables to zero. Let these variables be called C0 and C1.

3. For each octet of the TPDU, including the header, variable part of the header and the user data, add the octet value to C0, and then add the value of C0 to C1. Octets should be processed sequentially, starting with the first octet (the Length Indicator) and proceeding through the TPDU. All addition is to be performed modulo 255.

4. Calculate the value field of the checksum parameter as follows. Let the offset into the TPDU of the first octet of the value field be 'n' (where the first octet of the TPDU, the Length Indicator of the header, is considered to be at offset 1). Let the length of the TPDU, i.e. the number of times the above operation was repeated, be 'L'. Let the first octet of the checksum value, i.e., the one at offset 'n' be called 'X', and the second octet, at offset 'n+1', be called 'Y'. Then:

$$X = (((L - n) * C0) - C1) \text{ modulo } 255$$
$$Y = (((L - n + 1) * (-C0)) + C1) \text{ modulo } 255$$

5. Place the computed values of X and Y in the appropriate octets of the TPDU.

NOTE

> An implementation may use one's complete arithmetic as an alternative to modulo 255 arithmetic. However, if either of the checksum octets X and Y has the value minus zero (i.e., 255) then it must be converted to plus zero (i.e., 0) before being stored.

When a TPDU is received for a TC for which the checksum option has been selected, the TPDU must be verified to ensure that it has been received correctly. This is done by computing the checksum, using the same algorithm by which it was generated. The nature of the checksum algorithm is such that it is not necessary to compare explicitly the stored checksum bytes. The procedure described below may be used to verify that a TPDU has been correctly received.

1. Initialize two variable to zero. Let these variables be called C0 and C1.

2. For each octet in the received TPDU, add the value of the octet to C0 and then add the value of C0 to C1, starting with the first octet and proceeding sequentially through the TPDU. All addition is to be performed modulo 255.

3. When all octets have been sequentially processed, the values of C0 and C1 should be zero. If either or both of them is non-zero, the TPDU has been received incorrectly and the verification has failed. Otherwise, the TPDU has been received correctly and the TPDU should be processed normally.

NOTE

> An implementation may use one's complement arithmetic as an alternative to modulo 255 arithmetic. In this case, if either C0 or C1 has the value minus zero (i.e., 255) it is to be regarded as though it was plus zero (i.e., 0)

If a checksum verification failure occurs, it is not possible to determine the TC that the TPDU relates to, since the Reference field of the TPDU may have been received incorrectly. Therefore, all TCs multiplexed onto the same NC must be treated as though a network signalled error has occurred.

6.19 Frozen References

Purpose: To prevent re-use of a reference while TPDUs associated with the old use of the reference may still exist.

Description: When a transport entity determines that a particular connection has terminated, the reference shall be placed in a frozen state during which time it will not be reused. The circumstances under which this is done, and the period of time for which the reference remains frozen depends on the class.

6.20 Retransmission on Timeout

Purpose: To cope with unsignalled loss of TPDUs by the network service provider.

TPDUs and Fields Used:

CR, CC, DR, DT, ED, AK

Description:

The description is given in the section related to class 4.

6.21 Resequencing

Purpose: To cope with misordering of TPDUs by the network service provider.

TPDUs and Field Used:

DT
- TPDU NR

ED
- ED TPDU NR

Description:

The description is given in the section related to class 4.

6.22 Inactivity Control

Purpose: To cope with unsignalled termination of a network connection.

TPDUs and Fields Used:

AK

Description:

The description is given in the section related to class 4.

6.23 Treatment of Protocol Errors

Purpose: To deal with invalid TPDUs.

TPDUs and Fields Used:

ERR
- reject cause
- TPDU in error (string of octets)

DR
- reason code

Description:

This function is inherent.

Any received TPDU which is invalid or which cannot be dealt with by any operative function, or which is regarded as a violation of the protocol rules of the class in use (e.g., receipt in a wrong state, window error, sequencing error, TPDU with incorrect format), shall be considered as a protocol error. Such an error shall be signalled to the transport entity responsible by the sending of an TPDU Error (ERR) TPDU or by initiating a release. The ERR TPDU conveys the octets of the offending TPDU up to and including the octet where the error was detected.

In general, no further action is defined for the sender of ERR TPDU, since it is expected that the offender will either correct the error, or close the connection.

Action to be done by the receiver depends on local implementation decision; e.g., freeze the connection, report to management, disconnect.

NOTES:

1. Further action is a local implementation issue. Care should be taken by the transport entity receiving several invalid TPDUs or ERR TPDUs to avoid looping if the error is repeatedly generated.

2. There are two cases in which specific action is defined for the receiver of the ERR TPDU (see Sections 6.6 and 7.0.7).

6.24 Splitting and Recombining

Purpose: To allow a transport connection to make use of multiple network connections to provide additional resilience against network failure, to increase throughput, or for other reasons.

Description:

This function is available only in Class 4.

When this function is being used, a transport connection is assigned (see Section 6.1) to multiple network connections. TPDUs for the connection may be sent over any assigned network connection. The resequencing function of Class 4 (see Section 6.21) is used to ensure that TPDUs are processed in the correct sequence.

If the use of Class 4 is not accepted by the remote transport entity following the negotiation rules, only the network connection over which the CR TPDU was sent may be used for this transport connection.

The splitting function should only be used where the supporting network connections provide similar transmit delay.

Protocol Mechanism	Variant	0	1	2	3	4
Assignment to Network Conn.		*	*	*	*	*
TPDU Transfer		*	*	*	*	*
DT TPDU Length and Segmenting		*	*	*	*	*
Concatenation and Separation			*	*	*	*
Connection Establishment		*	*	*	*	*
Connection Refusal		*	*	*	*	*
Release	implicit	*				
	explicit		*	*	*	*
Implicit Termination		*		*		
DT TPDU Numbering	normal		*	m	m	m
	extended			(1)o	o	o
Expedited Data Transfer	network exp.		ao			
	not "		m	* (1)	*	*
Reassigment			*		*	
Reassignment after Failure			*		*	
Retention until Acknowledge-ment of TPDUs	Conf. Receipt		ao			
	AK		m		*	*
Resynchronization			*		*	
Multiplexing and Demultiplexing				*	*	*

Explicit Flow Control With			m	*	*
Without	*	*	o		
Checksum (use of)					m
(non-use of)	*	*	*	*	o
Frozen References					*
Retransmission on Timeout					*
Resequencing					*
Inactivity Control					*
Treatment of Protocol Errors	*	*	*	*	*
Splitting and recombining					*

(1) not applicable in class 2 when the non use of explicit flow control is selected.

7. PROTOCOL CLASSES

The details of the implementation of the protocol mechanisms are in certain cases different for different classes. For this reason, the following table is not intended to provide a complete description of the classes, but more to give an overview of how each class works. The exact definition of the protocol is given in the subsequent sections.

KEY

* include in the class (always)

m mandatory function (negotiable but always implemented)

o additional function (negotiable but not necessarily implemented)

ao additional function (negotiable but not necessarily implemented). Use of this option depends on the willingness of both transport entities and availability of network service.

na not applicable.

7.0 PROTOCOL DESCRIPTION OF CLASS 0: SIMPLE CLASS

7.0.1 Characteristics of Class 0

The characteristic of this class is that it provides the simplest type of transport connection and fully compatible with the CCITT recommendations S.70 for Teletex terminals.

The class is designed for use in association with network connections of type A (see 5.3.1.2.4.).

7.0.2 Functions of Class 0

This class is designed to have minimum functionality. It provides only the functions needed for connection establishment with negotiation, data transfer with segmenting and protocol error reporting.

Class 0 provides transport connections with flow control based on the network service provided flow control, and disconnection based on the network service disconnection.

7.0.3 Protocol Mechanisms of Class 0

7.0.3.1 Connection Establishment Phase

Connection shall be made in accordance with the general rules (Assignment of Network Connection, Connection Establishment and Connection Refusal) with the following restrictions:

- o No exchange of user data is allowed.
- o Only TSAP-ID and TPDU size parameters are allowed.

7.0.3.2 Data Transfer Phase

- o Segmenting (DT TDPU length and Segmenting)
- o Detection and indication of procedural errors.

7.0.3.3 Release Phase

There is no explicit transport connection release procedure for this class. The lifetime of the transport connection is directly correlated to the lifetime of the network connection.

7.0.4 Connection Establishment for Class 0

The connection establishment function is used with the contraint that only the transport entity which has requested the establishment of the network connection may send the CR TPDU. If the calling transport entity receives a CR TPDU, it shall transfer a TPDU Error (ERR) TPDU to notify the called transport entity of the procedure error.

7.0.5 Data Transfer Procedures

7.0.5.1 General

The data transfer procedures described in the following subsections apply only when the transport layer is in the data transfer phase, that is after completion of Transport Connection establishment.

7.0.5.2 Transport Data TPDU maximum length

For Class 0 the standard maximum transport data TPDU length is 128 octets including the data TPDU header octets.

Other maximum TPDU lengths may be supported in conjunction with the optional transport data TPDU size negotiation function (see Section 8.3 and 8.4). Optional maximum data field lengths shall be chosen from the following list: 256, 512, 1024 and 2048 octets.

TSDUs are transmitted using the segmenting function.

7.0.6 Release Procedure

The "implicit" variant of the release function is used.

7.0.7 Treatment of invalid TPDUs

The "treatment of protocol errors’ function is used.

7.0.8 Behaviour after an error signalled by the network service.

The implicit termination function is used and the high layer is informed about this disconnection.

7.0.9 Supported Options

None

7.1 PROTOCOL DESCRIPTION OF CLASS 1: BASIC ERROR RECOVERY CLASS

7.1.1 Characteristics of Class 1

The characteristic of this class is that it provides a basic transport connection with minimal overheads.

The main purpose of the class if to recover from network signalled errors (network disconnect or reset).

Selection of this class is usually based on reliability criteria. Class 1 has been designed to be used in association with type B network connections.

7.1.2 Functions of Class 1

Class 1 provides transport connections with flow control based on the network service provided flow control, error recovery, expedited data transfer, disconnection, and also the ability to support consecutive Transport connections on a network connection.

This class provides the functionality of Class 0 plus the ability to recover after a failure signalled by the Network Service, without involving the user of the Transport Service.

7.1.3 Protocol Mechanisms of Class 1

Class 1 protocol mechanisms include Class 0 protocol mechanisms plus the following:

7.1.3.1 User Data in the Connection Phase

Class 1 provides the possibility of conveying data in the connection request and confirm commands.

7.1.3.2 Numbering of Data TPDU

Each Data TPDU transmitted between transport entities for each direction of transmission in a transport connection is sequentially numbered.

7.1.3.3 Release

The "explicit" variant of the release function is used.

7.1.3.4 Error Recovery

The sending Transport entity keeps a copy of transmitted TPDUs until it receives an acknowledgment which allows copies to be released.
After a failure is indicated by the nerwork service (Reset, Disconnect), the resynchronization function is used to determine which TPDUs must be retransmitted.

Resynchronization may also be invoked by a transport entity as a local matter. For that purpose the Resynchronization function is used (see note at the end of Section 6.15).

7.1.3.5 Acknowledgement

Acknowledgements are used to release copies of retained TPDUs.

Two methods of acknowledgment are provided in the Retention until Acknowledgement of TPDUs function:

- o use of AK TPDU ("AK" variant) - mandatory

  Note: The credit field of the AK TPDU is not used in this class (always Set to zero).

- o use of network layer Confirmation of Receipt Service. ('confirmation of receipt' variant) - optional

The variant to be used is negotiated during the Connection Establishment Phase. The default option is the "AK TPDU" variant. Use of Network Layer Receipt Confirmation is allowed only in Class 1, and depends on the availability of the network layer receipt confirmation service, the expected cost reduction, and the agreement of both transport entities to use it.

7.1.4 Connection Establishment Procedures for Class 1

The 'assignment to network connection' and 'connection establishment' mechanisms are used. From the point at which a transport entity issues a CR proposing the use of Class 1 or a CC accepting the use of Class 1 the following mechanisms must be available to deal with signalled errors during connection establishment:

- o Reassignment after failure
- o Retention until Acknowledgement of TPDUs
- o Resynchronization

If no DT or ED TPDU is to be sent, receipt of a CC should be acknowledged.

7.1.5 Data Transfer Phase

Data transfer is accomplished using the 'TPDU transfer' 'Concatenation' and 'DT TPDU Length and Segmenting' mechanisms. 'DT TPDU Numbering' and 'Retention until Acknowledgement of TPDUs' are used in support of error recovery.

7.1.5.1 Behaviour after an error

After receiving a network reset, the Resynchronization mechanism is invoked. After receiving a network disconnect, the 'Reassignment after Failure' mechanism is invoked after which the 'Resynchronization' mechanism is invoked.

The 'Spurious Disconnect' mechanism is used to deal with receipt of a DR TPDU for an unrecognised Transport

Connection.

7.1.5.2 Procedure for Expedited Data Transfer

The Expedited Data Transfer mechanism is used. Two methods are possible to provide the function:

o non network expedited variant

Note: (1) This method is always included in this class.

Note: (2) The EDTPDU-NR of the ED TPDU contains an identification number. This number must be different for successive ED TPDU s. That is, when an ED TPDU has been sent and an EA TPDU for the ED TPDU has been received, the next ED TPDU must have a different value in the EDTPDU-NR field. No other significance is attached to EDTPDU-NR field. It is recommended but not essential, that the values used be consecutive modulo 128.

o network expedited variant

Note: (1) The use of this method is determined through negotiation during transport connection establishment.

7.1.6 Release Procedures

The 'explicit' variant of the Release mechanism is used.

Receipt of an error indication by a transport entity, which, prior to this event has sent a DR, causes this transport entity to retransmit DR. Only DC and DR will be accepted and interpreted as the completion of the connection release sequence. The related Reference will become unassigned.

7.1.7 Treatment of Unknown TPDUs

The 'Treatment of Protocol Errors' mechanism is used.

7.1.8 Supported Options

Use of network receipt confirmation.

Use of network expedited.

7.2 PROTOCOL DESCRIPTION OF CLASS 2: MULTIPLEXING CLASS

7.2.1 Characteristics of Class 2

The characteristic of this class is to provide a way to multiplex several transport connections onto a single network connection. This class has been designed to be used in association with type A network connections.

Use of Explicit Flow Control

The objective is to provide flow control to help avoid congestion at end-points and on the network connection. Typical use is when traffic is heavy and continuous, or when there is intensive multiplexing. Use of flow control can optimize response times and resource utilization.

Non Use of Explicit Flow Control (optional)

The objective is to provide a basic transport connection with minimal overheads suitable when independence of transport and network connection lifetime is desirable. The class would typically be used for unsophisticated terminals, and when no multiplexing onto network connections is required. Expedited data is never available.

7.2.2 Functions of Class 2

Class 2 provides transport connections with or without individual flow control - no error detection or error recovery is provided.

If the network resets or clears, the transport connection is terminated without the transport clearing sequence and the transport user is informed.

When explicit flow control is used a credit mechanism is defined allowing the receiver to inform the sender of the exact amount of data he is willing to receive and expedited data transfer is available.

7.2.3 Protocol Mechanisms of Class 2

7.2.3.1 Connection Establishment Phase

The connection establishment function shall be used.

7.2.3.1.1 User Data in the Connection Phase

Class 2 provides the possibility to convey data in the connection request and confirm commands.

7.2.3.2 Connection Identification

In Class 2 each TPDU conveys a Destination Reference.

This uniquely identifies the transport connection within the receiving transport entity and thus allows multiplexing.

7.2.3.3 Release Phase

The release of a transport connection results either from the use of the 'explicit' variant of the release function or from the Implicit Termination function.

7.2.3.4 Protocol Mechanisms when Explicit Flow Control is used.

The following mechanisms are provided:

7.2.3.4.1 Numbering of Data TPDU

Each Data TPDU transmitted between transport entities for each direction of transmission in a transport connection is sequentially numbered.

Each Data TPDU contains a Send Sequence Number T(S).

7.2.3.4.2 Flow Control Principles

The receiver of data TPDUs holds a count of the sequence number of the next expected TPDU. This count is called the Receive Sequence Number, T(R). The receiver indicated to the sender the number of Data TPDUs he is ready to receive by means of a 'credit' mechanism. Credits are given using the credit field in the AK TPDU. The value of the credit field, in conjunction with the value of T(R) transported by the YR-TU-NR (your TPDU number) field of the AK TPDU, is used by the receiver of the AK TPDU to determine whether and how many Data TPDUs may be accepted by the sender of the AK TPDU. Precise definition of flow control principles appears in Section 7.2.5.5.3.

7.2.3.4.3 Expedited Flow

The non network expedited variant is used. Normal flow is the flow of data subject to the flow control mechanism, expedited flow is the flow of data that the sender may send without explicit agreement of the receiver. This expedited flow has a limited capability and could for example be used to carry session supervisory commands.

The number of expedited data units outstanding at any time is limited to one and the amount of TS-user data is limited (up to 16 octets).

An expedited data may arrive before normal data which was submitted earlier. Normal data submitted after the expedited data will arrive after the expedited data.

7.2.4 Connection Establishment Procedures for Class 2

7.2.4.1 References

See Section 6.5 for reference assignment. Receipt of any TPDU with a reference that is not assigned to a transport connection other than a Disconnect Request (DR) or Connection Request (CR) will be ignored.

Receipt of a Disconnect Request (DR) for an unassigned Reference will result in a Disconnect Confirm (DC) response.

7.2.4.2 Connection Eastablishment

This phase is achieved by exchange of CR/CC TPDU using the 'connection establishment' function. Since the multiplexing function is in use, then more than one transport connection may be assigned to the same network connection concurrently. The restrictions of Class 0 does not apply to this class and the other higher classes.

7.2.5 Data Transfer Procedures for Class 2

The data transfer procedures described in the following section apply independently to each transport connection existing between two transport entities.

7.2.5.1 TPDU Maximum Length and Segmenting

The general rules defined in Section 6.3 apply.

7.2.5.2 Concatenation

The general rules defined in Section 6.4 apply.

7.2.5.3 Sending Data TPDU (No Explicit Flow Control Option)

In this case the data TPDU is built in accordance with the rules stated in Section 6.2 and 6.3 and sent without any additional mechanisms. Thus, the DT TPDU NR field may take any value and no AK TPDU is used.

7.2.5.4 Sending Data TPDU (When Explicit Flow Control is Used)

On each transport connection the transmission of Data TPDUs is controlled separately for each direction and is based on authorization from the receiver.

This authorization is provided through the use of the TPDUs Credit field. Credit field values are only present in the following TPDUs: CR, CC, AK..

7.2.5.4.1 Numbering of Data TPDUs

Each Data TPDU transmitted between transport entities, for each direction of transmission in a transport connection, is sequentially numbered.

The sender of Data TPDUs holds a count of the next TPDU to be sent. This count is called the Send Sequence Number T(S). The sender indicates to the receiver the number of the data TPDU he sends by putting the current T(S) value into the TPDU-NR field of the data TPDU.

Sequence numbering is performed modulo 2**n, where n is the number of bits of the sequence number field. The T(S) counter cycles through the entire range 0 to (2**n)-1.

At connection establishment time both Transport entities initialize their T(S) and T(R) counts to zero (i.e. the first Data TPDU to be transmitted between transport entities for a given direction of data transmission after the connection establishment has a TPDU-NR field set to zero).

Receipt of a Data TPDU whose TPDU-NR field is not equal to the expected value T(R), is to be regarded as a protocol error.

Operations described above are summarized as follows:

- initalization

  T(S) = 0   T(R) = 0

  Sending of Data TPDU

  put T(S) into the TPDU-NR field of the Data TPDU to be sent

  T(S) = (T(S) + 1) (modulo 2**n)

  Receiving of Data TPDU

  TPDU-NR field of the received data TPDU which is not equal to T(R) is a protocol error.

  T(R) = (T(R) + 1) (modulo 2**n)

7.2.5.4.2 Window Definition

For each transport connection and for each direction of data transmission a 'transmit window' is defined as the (possibly null) ordered set of consecutive data TPDUs authorized to be transmitted in that direction. At any given time, the lowest sequence number of a data TPDU which a transport entity is authorized to transmit is referred to as the 'lowest window edge'. The 'upper window edge' is calculated by adding the credit allocation, given by the value of the Credit (CDT) field contained in a received TPDU, to the lower window edge. Note that a transport entity is authorized to send data TPDUs with sequence numbers up to but not including the upper window edge.

7.2.5.4.3 Flow Control

Flow control is performed as follows:

o initialization time

Lower window edge = 0

Upper window edge = N (Credit received either in CR or in CC and N < 2**p < 2** (n-1), where P is the number of bits in credit field of CR and CC.

o Sending of a Data TPDU

Send data TPDUs while T(S) is less than the upper window edge. If T(S) equals the upper window edge then wait for additional credit before sending.

o Reception of Data TPDU (with TPDU NR = T(R)

If T(R) is greater than or equal to the upper window edge authorized to the sending transport entity, then the receiving transport entity shall use the Treatment of Protocol Errors function. Otherwise T(R) shall be incremented.

Sending Credit

Send AK TPDU with YR-TU-NR = T(R) and Credit equals N. (Where N = number of additional data TPDUs the entity is prepared to receive.)

Receiving Credit in AK.

Lower window edge = YR-TU-NR received.

Upper window edge = Lower window edge + N.

7.2.5.4.4 Reducing the Upper Window Edge

The value of the upper window edge cannot be decreased in this class. If, at a certain point of time, the upper window edge value is U, the reception of an AK TPDU having YR-TU-NR = M and CDT = N such that:

(U-M) (mod. 2**n) > N

is a protocol error

Provided the previous statements are respected, CDT field may take any value including zero.

7.2.5.4.5 Procedure for Expedited Data Transfer

The procedure of expedited data transfer allows a transport entity to transmit data to the remote transport entity without following the flow control procedure of the normal data flow. This procedure can only apply in the transfer phase.

The expedited procedure has no effect on the transfer and flow control applying to normal Data TPDUs.

To transmit expedited data, the transport entity sends an expedited data TPDU (ED TPDU). The size of a data field is limited (up to 16 octets). The data field contains a complete ED TSDU. The remote transport will then confirm the receipt of the ED TPDU by transmitting an expedited TPDU acknowledgement (EA TPDU). A transport entity can send another ED TPDU only after having received an EA TPDU for the previously transmitted ED TPDU. In class 2 the ED TPDU NR field of the ED and YR-TU-NR field of the EA TPDU are not defined and may take any value.

7.2.6 Release Procedures for Class 2

The data phase ends after a transport entity has sent or received a Disconnect Request (DR). The transport entity will ignore any incoming TPDU except DC or DR.

If the network resets or clears the network connection, all transport connections are terminated without the transport clearing sequence. The References become frozen.

For Class 2 the explicit variant of the ’release’ mechanism is used, enabling transport connections to be cleared independently of the underlying network connection.

7.2.7 Treatment of Invalid TPDUs

The 'Treatment of Protocol Error' mechanism in Section 6.23 is used.

7.2.8 Behaviour after an Error signalled by the Network Layer.

The implicit termination mechanism is used.

7.2.9 Supported Options

Non use of explicit flow control.
Extended formats.

7.3 PROTOCOL DESCRIPTION OF CLASS 3: ERROR RECOVERY AND MULTIPLEXING CLASS

7.3.1 Characteristics of Class 3

The characteristics of Class 3 in addition to those of Class 2 is to mask errors indicated by the network. Selection of this class is usually based upon reliability criteria. Class 3 has been designed to be used in association with type B network connections.

7.3.2 Functions of Class 3

This class provides the functionality of Class 2 (with use of explicit flow control) plus the ability to recover after a failure signalled by the Network Layer without involving the user of the transport service.

The mechanisms used to achieve this functionality also allow the implementation of more flexible flow control.

7.3.3 Protocol Mechanisms of Class 3

Class 3 mechanisms include Class 2 (with use of explicit flow control option) mechanisms and the ability to recover after a failure signalled by the network without informing the user of the transport connection.

7.3.3.1 Error Recovery Principles

The sending transport entity keeps a copy of transmitted Data TPDUs and ED TPDUs until it receives a positive aknowledgement which allows copies to be released. It may also receive an RJ command inviting it to retransmit or transmit all Data TPDUs, if any, from the point in the sequence indicated in the RJ command.

This is especially the case, when a failure is indicated by the network. The transport entity sends an RJ command in order to indicate the sequence number of the next expected TPDU.

Error recovery for ED TPDU is achieved by retransmission (see 7.3.5.3).

7.3.3.2 Relationship between Flow Control and Error Recovery

Acknowledgement is performed by use of the T(R) count. A credit is associated with this acknowledgement which may be equal to or greater than zero. Thus it is possible to acknowledge data without giving the right to send new data.

Credit may be reduced, by the use of the RJ TPDU.

7.3.4 Connection Establishment Procedure for Class 3

The rules for Class 2 (with use of explicit flow control) apply with the addition of the following rules which apply on receipt of an eror indication from the Network layer.

- Reception of an error indication by a transport entity which, prior to this event, has sent a CR and has not yer received a CC, causes the transport entity to retransmit CR.

- Reception of an error indication by a transport entity to wait for reception of CR, RJ or DR TPDU. In this case:

  - Reception of CR will cause the transport entity to retransmit CC.

  - Reception of RJ will cause the transport entity to transmit an RJ with a YR-TU-NR equal to zero and enter the data phase.

  - Reception of a DR will cause termination of the transport connection as for Classes 1 and 2 (see 7.1.4).

7.3.5 Data Transfer Procedures for Class 3

7.3.5.1 Acknowledgement

The 'AK' variant of the Retention until Acknowledgement of TPDUs function is used.

7.3.5.2 Retransmission Procedure

TPDU retransmission is a procedure which allows a transport entity to request retransmission of one or several consecutive Data TPDUs from the remote transport entity. A transport reject condition is signalled to the remote transport entity by transmission of an RJ TPDU whose YR-TU-NR field indicates the sequence number of the next expected Data TPDU.

On receipt of a RJ TPDU, a Transport entity shall accept credit to the value contained in the credit field and shall re-transmit TPDUs, starting with the one whose number is specified in the YR-TU-NR field of the received RJ TPDU, subject to the new credit.

The transport entity shall not specify a T(R) in the RJ TPDU less than that which has previously been acknowledged. Receipt of an RJ TPDU with a T(R) which has been previously acknowledged will be considered a protocol error.

Additional DT TPDUs pending initial transmission may follow the retransmitted DT TPDU(s) if the window is not closed.

7.3.5.3 Reducing the upper window edge

It is possible to decrease the value of the upper window edge down to the sequence number transported by YR-TU-NR field of the RJ TPDU. Receipt of an DT TPDU which would have been inside the window before the reduction is not a protocol error and this TPDU may be discarded.

Note: In such a case the credit equal to zero achieves the effect of a Receive not Ready Condition.

7.3.5.4 Behaviour after an error signalled by the network layer

After receiving an error indication from the Network Service, the transport entity shall tranmit to the remove entity an RJ TPDU with YR-TU-NR field indicating the sequence number of the next expected Data TPDU.

7.3.5.5 Procedure for Expedited Data Transfer

In Class 3, the ED TPDU-NR field of the Expedited Data (ED) TPDU contains an identification number. This number must be different for successive ED TPDUs. That is, when an ED TPDU has been sent and an EA TPDU for the ED TPDU has been received, the next ED TPDU must have a different value in the NR field of the ED TPDU. No other significance is attached to this field. It is recommended, however, that the values used be consecutive modulo 2**n. When a transport entity receives an ED TPDU for a transport connection, it shall respond by transmitting an expedited acknowledgement (EA) TPDU.

It places in the YR-TU-NR field the value contained in the NR field of the received ED TPDU. If, and only if, this value is different from the NR field of the previously received ED TPDU, the data contained in the TPDU is to be passed to the session entity.

If an error indication from the Network layer is received before the receipt of the expected Expedited Acknowledgement (EA) TPDU, the transport entity shall retransmit the ED TPDU with the same value in the NR field. By the rule described in the previous paragraph, the session entity does not receive data corresponding to the same expedited TPDU more than once.

7.3.6 Release Procedures for Class 3

The rules for Class 2 apply with the addition of the following rule:

Receipt of an eror indication by a transport entity, which prior to this event has sent a DR, causes this transport entity to retransmit DR. Only DC and DR will be accepted and interpreted as the completion of the connection clearing sequence. The related Reference will become unassigned.

7.3.7 Treatment of Invalid TPDUs

The 'Treatment of Protocol Errors' mechanism is used.

7.3.8 Supported Options

Extended formats.

7.4 PROTOCOL DESCRIPTION OF CLASS 4: ERROR DETECTION AND RECOVERY CLASS

7.4.1 Characteristics of Class 4

The characteristic of Class 4, in addition to those of Class 3, is the detection of errors which occur as a result of the low grade of service available from the network layer. The kinds of errors to be detected include: TPDU loss, TPDU delivery out of sequence, TPDU duplication. These errors may afect control TPDUs as well as Data TPDUs.

Class 4 has been designed to be usd in association with network connections of type C.

7.4.2 Functions of Class 4

This class provides the functionality of Class 3, plus the ability to detect and recover from lost, duplicated or out of sequence TPDUs without involving the user of the transport service.

This detection of errors is made by extended use of the sequence numbering of Classes 2 and 3, by a timeout mechanism, and by additional protocol mechanisms.

This class additionally detects and recovers from damaged TPDUs by using a checksum mechamism. The use of the checksum mechanism must be available but its use or its non use is subject to negotiation. Class 4 does not attempt to deal with detection of errors due to the misdelivery of TPDUs.

7.4.3 Protocol Mechanisms of Class 4

7.4.3.1 Network Service Data Unit Lifetime

The network layer is assumed to provide, as an aspect of its grade of service, for a bound on the maximum lifetime of NSDUs in the network. This value is known by the Transport Layer. The maximum time which may elapse between the transmission of an NSDU into the network layer and the receipt of any copy of it is referred to as M.

7.4.3.2 Average Transit Delay

It is assumed that there is some value of transmit delay in the network, typically much less than M, which will be the maximum delay suffered by all but a small proportion of NSDUs. This value is referred to as E.

7.4.3.3 Remote Acknowledge Time Assumptions

Any transport entity is assumed to provide a bound for the maximum time which can elapse between its receipt of a TPDU from the Network Layer and its transmisssion of the Corresponding response. this value is referred to as A/L. The corresponding time given by the remote transport entity is referred to as A/R. The values for these timers may be conventionally established or may be established at connection establishment time.

7.4.3.4 Local Retransmission Time

The local transport entity is assumed to maintain a bound on the time it will wait for an acknowledgement before retransmitting the TPDU. This time is the local retransmission time and is referred to as T1.

$$T1 = 2*E + X + Ar?$$

Where X is a value to allow for TPDU processing in the local transport entity.

7.4.3.5 Persistence Time

The local transport entity is assumed to provide a bound for the maximum time for which it may continue to retransmit a TPDU requiring positive acknowledgment. This value is referred to as R.

The value is clearly related to the time elapsed between retransmission, T1, and the maximum number of retransmissions, N. It is not less than T1*N+X, where X is small quantity to allow for additional internal delays, the granularity of the mechanism used to implement T1 and so on. Because R is a bound, the exact value of X is unimportant as long as it is bounded and the value of a bound is known.

7.4.3.6 Bound on Reference Identifier and Sequence Numbers

Using the above values, a bound L may be established for the maximum time between the decision to transmit a TPDU and the receipt of any response relating to it. The value of L is given by:

$$L = 2*M+R+Ar$$

It is necessary to wait for a period L before reusing any reference or sequence number, to avoid confusion in case a TPDU referring to it may be duplicated or delayed.

(Note: In practive, the value of L may be unacceptably large. It may also be only a statistical figure at a certain confidence level. A smaller value may therefore be used where this still allows the required quality of service to be provided).

7.4.3.7 Inactivity Time

To protect against unsignalled breaks in the network connection (Half-open connections), each transport entity maintains an inactivity time interval. If the interval passes without receipt of some TPDU, the transport entity will terminate the TC by making use of the release procedure. This interval is referred to as I.

7.4.3.8 Window Time

A transport entity maintains a time to ensure that there is a maximum interval between transmission of up-to-date window information. This interval is referred to as the window time, W.

7.4.3.9 Class 4 Error Recovery Principles

In class 4, the transport entity associates a response time with TPDUs sent requiring a response. If an appropriate response is not received within time T1, the recovery procedure must be invoked by the sender. This will usually involve the retransmission of the corresponding TPDU.

A TPDU may be transmitted a maximum number of times, This number is referred to a N. The value of N is chosen so that the required quality of service can be provided given the known characteristics of the network connection.

7.4.3.10 Relationship of Times and Intervals

The following note describes the relationship between the time described in Section 7.4.3.1 - 7.4.3.9.

Note:

a. The interrelationship of times for the worst case is as follows:

M: maximum transit delay of the network (see 7.4.3.1)

Ar maximum acknowledgement time of the remote transport entity (see 7.4.3.3)

R: maximum local retransmission time (see 7.4.3.5)

N: maximum number of transmission for a single TPDU (see 7.4.3.9)

L: maximum time for a TPDU to be valid (see 7.4.3.6)

$$R = T_1 * (N-1)$$

R

*

M

L *

A_R $= 2*M + A_R + R$

*

M

t t

b. The interrelationship of times for the average case is as follows (see 7.4.3.4)

E: average transit delay for the network (E<<M)

X: TPDU processing time

T_1: average time from sending a TPDU until the receipt of its acknowledgement (see 7.4.3.4)

A_R: maximum acknowledgement time of the remote transport entity (see 7.4.3.3)

X

E

A_R $T_1 = 2*E + X + A_R$

E

t t

7.4.3.11 Sequence Numbering

In Class 4 sequence numbering is applied to certain control TPDUs and their acknowledgements, as well as to DT TPDUs. These are ED and its acknowledgement EA.

The length of sequence numbers may be negotiated at connection establishment. Where other than the default length is used, an extended header format is used for sequenced TPDUs containing additional octets of sequence numbers. Extended header format includes a credit field on the appropriate TPDU types allowing extended credit allocation.

7.4.4 Procedures for Connection Establishment Phase

The following features pertain to connection establishment for Class 4:

o In Class 4, a connection is not considered established until the successful completion of a 3-way TPDU exchange. The sender of a CR TPDU must respond to the corresponding CC

TPDU by immediately sending a DT, ED or AK TPDU.

- o As a result of duplication, a CR TPDU may be received specifying a source reference which is already in use with the sending transport entity. If the receiving transport entity is in the data transfer phase, having completed the 3-way TPDU exchange procedure, the receiving transport entity should ignore such a TPDU. Otherwise a CC TPDU should be transmitted.

- o As a result of duplication or retransmission, a CC TPDU may be received specifying a paired reference which is already in use. The receiving transport entity should ignore such a CC TPDU.

- o A CC TPDU may be received specifying a reference which is in the frozen state. The response to such a TPDU should be a DR TPDU.

7.4.4.1 Connection Request

When a transport entity transmits a CR TPDU it starts timer T1. If this timer expires before a CC TPDU is received, the CR TPDU is retransmitted and the timer restarted. After transmission of the CR TPDU N times, the connection establishment procedure is abandoned and the failure reported to the transport user. The reference must be placed in the frozen state for a period L (see section 7.4.3.6).

7.4.4.2 Incomimg Connection Request

An incoming connection request is processed as for Class 3

7.4.4.3 Connection Confirm

When a transport entity transmits a CC TPDU it starts timer T1. If this timer expires before an AK or DT TPDU is received, the CC TPDU is retransmitted according to the retransmission principles in Section 7.4.3.9

7.4.4.4 Incoming Connection Confirm

When a CC TPDU is received, the receiving transport entity enters the data transfer phase. It must immediately transmit an AK, ED or DT TPDU to complete the 3-way TPDU exchange. The CC TPDU is subject to the usual rules of the data transfer phase regarding retransmission, see Section 7.4.5.3.

7.4.4.5 Incoming Acknowledgement

When an AK, DT or ED TPDU is received the receiving transport entity can enter the data transfer phase. If the entity has data to send it may send DT TPDUs or an ED TPDU. The DT TPDUs are subject to flow control. Otherwise, the transport entity must obey the inactivity principles (see Section 7.4.5.8).

7.4.4.6 Unsuccessful Connection

When a DR TPDU is received in response to a CR TPDU, the timer T1 is cancelled and the reference placed in the frozen state for a period L (see Section 7.4.6.1).

7.4.4.7 Initial Credit Allocation

The CR and CC TPDUs may allocate an initial credit value to their respective recipients. This value is limited to 15 by the encoding of the TPDU. Where the extended header format is in use, credit values greater than 15 must be allocated using AK TPDUs.

7.4.4.8 Exchange of Acknowledge Time

A transport entity may transmit the value it intends to use for AL at connection establishment, as the 'Acknowledge Time' parameter in the CR or CC TPDU (depending on whether the transport entity is initiating or accepting the transport connection). If this parameter is present in a received CR or CC TPDU, the value of AR should be set accordingly. If this parameter is not present, AR may be assumed to be insignificant in comparison to E the typical maximum transit delay.

7.4.5 Procedure for Data Transfer Phase

7.4.5.1 Sequence Control

The receiving transport entity is responsible for maintaining the proper sequence of DT TPDUs.

DT TPDUs received out of sequence must not be delivered to the TS-user until in-sequence TPDUs have also been received.

AK TPDUs also contain information allowing the receiving transport entity to process them in the correct order.

7.4.5.2 Duplicate DT TPDUs

Duplicate TPDUs can be detected because the T(S) value matches that of previously received TPDUs. T(S) values must not be reused for the period L after their previous use. Otherwise, a new, valid TPDU could be confused with a duplicated TPDU which had previously been received and acknowledged.

Duplicated DT TPDUs must be acknowledged, since the duplicated TPDU may be the result of a retransmission resulting from the loss of an AK TPDU.

The data contained in a duplicated DT TPDU should be ignored.

7.4.5.3 Retransmission Principles

When a transport entity has some outstanding DT or ED TPDUs that require acknowledgement, it will check that no T1 interval elapses without the arrival of an AK or EA TPDU that acknowledges one of them. If the timer expires, the first TPDU is retransmitted and the timer is restarted. After N transmissions (N−1 retransmissions) the connection is assumed to have failed and the release phase is entered, and the transport user is informed.

DT TPDUs which fall beyond the current window (due to reduction of the upper window edge) are not retransmitted until advancement of the upper window edge so permits.

Note: This requirement can be met by different means, for example.

1. One timer is associated with each TPDU. If the timer expires, the associated TPDU will be retransmitted, and the timer T1 will be restarted for all subsequent DT TPDUs.

2. One timer is associated with each TC:

   if the transport entity transmits a DT TPDU requiring acknowledgement, it starts timer T1,

   if the transport entity receives a TPDU that acknowledges one of the TPDUs to be acknowledged, timer T1 is restarted,

   if the transport entity receives a TPDU that acknowledges the last TPDU to be acknowledged, timer T1 is stopped.

For the decision whether the retransmission timer T1 is maintained on a per TPDU or on a per TC basis, throughput considerations have to be taken into account.

7.4.5.4 Acknowledgement Principles

A transport entity operating class 4 must acknowledge all TPDUs received requiring acknowledgment. To avoid unnecessary retransmissions and to avoid delays to transmission by the remote transport entity, the delay for acknowledgement should not exceed timer A_L (see Section 7.4.3.2).

There are two TPDU types that must be acknowledged: ED and DT. Receipt of an ED TPDU must be acknowledged by an EA TPDU. A DT TPDU is acknowledged with an AK TPDU.

An AK TPDU has the sequence number of the next DT TPDU the receiving transport entity expects to receive. It thus acknowledges receipt of all DT TPDUs with sequence numbers less than the acknowledgement number.

An AK TPDU may be repeated at any time, using the sequence number in the last AK TPDU sent.

7.4.5.5 Flow Control Principles

Flow control in Class 4 is subject to the same principles as in Classes 2 and 3. The credit mechanism and window principle of those classes still apply, except that in class 4, the upper window edge can be reduced by the receiving transport entity by sending an AK TPDU with a smaller credit.

A receiving transport entity may send an AK TPDU at any time to change the window size. This AK TPDU may acknowledge a new DT TPDU or may repeat a previous acknowledgement.

7.4.5.6 Window Synchronization Principles

To ensure the synchronization of flow control information the window timer provokes the frequent exchange of AK TPDUs between transport entities. The window timer maintains a minimum level of TPDU traffic between transport entities cooperating in a transport connection.

In Class 4 the window size can be reduced in any AK TPDU. Due to the possibility of misordering of AK TPDUs and the associated loss of efficiency, the AK TPDU for class 4 includes an additional field called the AK TPDU subsequence parameter.

An AK TPDU should contain the subsequence parameter whenever a duplicate AK TPDU is sent with the same sequence number but with reduced credit. The value of the subsequence parameter is set to one for the first time the AK TPDU is resent with reduced credit.

When an AK TPDU is transmitted whose sequence number is increased, the 'sub-sequence' parameter is omitted until credit reduction takes place.

When an AK TPDU is received, it must be processed (i.e., its contents made use of) only if:

- The sequence number is greater than in any previously received AK TPDU, or,

- The sequence number is equal to the highest in any previously received AK TPDU, and the sub-sequence parameter is greater than in any previously received AK TPDU having the same sequence number (where an absent sub-sequence parameter is regarded as having a value of zero), or

- The sequence number and sub-sequence parameter are both equal to the highest in any previously received AK TPDU (where an absent sub-sequence parameter is regarded as having a value of zero), and the credit field is greater than in any previously received AK TPDU having the same sequence and sub-sequence numbers.

When an AK TPDU is transmitted which opens a closed window (i.e. increases credit from zero), it should be retransmitted at an interval of T1. Transmission should occur a maximum of N times, after which the usual inactivity retransmission timer should be reverted to. Retransmission may also cease if the local transport entity becomes sure that the new credit information has been received by the remote transport entity.

If a transport entity receives an AK TPDU containing a 'Flow Control Confirmation' parameter, whose Lower Window Edge and Your-Sub-Sequence fields are equal to its own lower window edge and sub-sequence number, it may note that the credit available at the remote transport entity (relative the Lower Window Edge field) is at least equal to the value conveyed as Your Credit. This enables the transport entity to cease the frequent retransmission of window information, if it thereby knows that the remote window is open.

A transport entity need not retransmit window information (except as described under Inactivity Principles) if it is aware by the receipt of DT TPDUs that the remote transport entity has sufficient credit to prevent deadlock. When an AK TPDU is transmitted in response to a DT TPDU, the transport entity may normally assume that the transmitter of the DT TPDU will ensure that the AK TPDU is received, be retransmission of the DT TPDU if necessary. Therefore, it can normally be assumed that the credit conveyed in such an AK TPDU will be available to the remote transport entity, and frequent retransmission is unnecessary.

The assumption that the DT TPDU will be retransmitted may be incorrect if credit reduction has taken place. Therefore, a transport entity may not make this assumption if the sequence number of the DT TPDU is less than or equal to the highest value for which permission to transmit (i.e., credit) has been given and subsequently withdrawn.

Upon receipt of an AK TPDU which increases the upper window edge, a transport entity may transmit an AK TPDU which repeats the information contained in the received TPDU in a 'Flow Control Confirmation' parameter in its variable part an thereby assures the transmitter of the original AK TPDU of its own state. Such an AK TPDU may be tranmmitted:

- Upon receipt of a duplicated AK TPDU (i.e., one which is identical in all fields, including the sub-sequence parameter if present, to the most recently received AK TPDU which was not discarded due to detection of a sequence error), not containing the 'Flow Control Confirmation' parameter.

- Upon receipt of an AK TPDU which increases the upper window edge but does not increase the lower window edge, when the upper window edge was formerly equal to the lower window edge.

7.4.5.7 Procedure for Expedited Data

The procedure for expedited data is as for Class 3, with the following exceptions.

The ED TPDU has a sequence number which is allocated from a separate sequence space from that of the DT TPDUs. The EA TPDU carries the same sequence number as the corresponding ED TPDU. Only a single ED TPDU may be transmitted and awaiting acknowledgements at any time.

Upon receipt of an unduplicated ED TPDU, a transport entity immediately forwards the data to the transport user. The ED

TPDU sequence number is recorded in an EA TPDU sent to the other transport entity.

The sender of an ED TPDU shall not send any new DT TPDU with higher T(S) until it receives the EA TPDU. This guarantees the arrival of the ED TPDU before any subsequently sent DT TPDUs.

7.4.5.8 Inactivity Principles

If the Inactivity Time I passes without receipt of some TPDU, the transport entity will terminate the TC by making use of the release procedure. To present expiration of the remote transport entity's inactivity times when no data is being sent, the local transport entity must send AK TPDUs at suitable intervals in the absence of data, having regard to the probability of TPDU loss. The Window Synchronization Principles (see 7.4.5.6) may ensure that this requirement is met.

Note: It is likely that the release procedure initiated due to inactivity timer expiration will fail, as such expiration indicates probable failure of the supporting NC or of the remote transport entity. This case is described in Section 7.4.6.

7.4.6 Procedures for Release Phase

The rules for class 3 apply. The DR TPDU is subject to the usual retransmission procedure. After N retransmissions, the transport connection is considered disconnected, the Reference is placed in the frozen state for a period L and retransmission ceases.

The DC TPDU is sent only in response to a DR TPDU, and is not subject to the retransmission procedure.

The DC TPDU when received allows the transport entity to release all resources maintained for the transport connection.

The DR TPDU does not carry a sequence number. Any previously transmitted TPDUs (including DT and ED) which are received after the DR TPDU result in a further DR TPDU but are otherwise ignored. After disconnection, for whatever reason, the Reference is placed in the frozen state for a period L.

7.4.6.1 Unassigned Frozen References

When a transport connection is terminated, the corresponding references cannot be immediately reused since TPDUs containing these references may continue to exist in the network for a time up to L. Therefore, these references will be placed in an unassignable, frozen state for this peiod.

After an event involving loss of transport entity state information, including the status of reference assignments, all references relating to network connections whose transport state information has been lost must be placed in the frozen state for a period L.

If a DC TPDU is received for a local reference which is in the frozen state, or with a remore reference not matching the already recorded one, this DC TPDU shall be ignored.

7.4.7 Treatment of Invalid TPDUs

The 'Treatment of Protocol Erorrs' function is used.

7.4.8 Supported Options

Non use of checksum.

Use of extended formats.

8. ENCODING

8.1 Summary

	Classes 0	1	2	3	4	Sect.	Code
CR Connection Request	x	x	x	x	x	8.3	1110xxxx
CC Connection Confirm	x	x	x	x	x	8.4	1101xxxx
DR Disconnect Request	x	x	x	x	x	8.5	10000000
DC Disconnect Confirm		x	x	x	x	8.6	11000000
DT Data	x	x	x	x	x	8.7	11110000
ED Expedited Data		x	NF	x	x	8.8	00010000
AK Data Acknowledgement (Note 1)		NRC	NF	x	x	8.9	0110xxxx
EA Expedited Data Acknowledgement		x	NF	x	x	8.10	00100000
RJ Reject (Note 1)		x		x		8.11	0101xxxx
ERR TPDU Error	x	x	x	x	x	8.12	01110000
not available (Note 2)						-	00000000

```
not available (Note 2)                            -      00110000

not available (Note 2)                            -      1001xxxx

not available (Note 2)                            -      1010xxxx
```

Where xxxx (bits 4-1) is used to signal the CDT

Note 1: In extended header format, the code for AK=0110 0000 and the code for RJ=0101 0000.

Note 2: These codes are already in use in compatible protocols defines by standards organizations other than CCITT/ISO.

NF: Not available when the non explicit flow control option is selected.

NRC: Not available when the receipt confirmation option is selected.

8.2 Structure

As defined in the previous sections, all the Transport Protocol Data Units (TPDU) shall contain an integral number of octets. The octets in a TPDU are numbered starting from 1 and increasing in the order of transmission. The bits in an octet are numbered from 1 to 8, where bit 1 is the low-ordered bit.

There are tao types of TPDUs:

- o Data TPDUs, used to transfer Transport Service Data Units (TSDUs). The structure of the TSDUs is maintained by means of the TSDU End Mark.

- o Control TPDUs, used to control the transport protocol functions, including the optional functions.

```
Octets  1  2  3  4          n  n+1         p  p+1
        ------------      --------------   --------------   --------
        LI|  | |  |  ...    |  |   |    .... | |    |   .... |
        ------------      --------------   --------------   --------

         <--- Fixed Part ----->&lt;-- Variable Part->
                                (including checksum
                                 where applicable)

        <--------------Header------------------->&lt;----Data Field->
```

A TPDU is divided into four parts:

- o Length Indicator Field (LI)
- o Fixed Part
- o Variable Part (may be omitted)
- o Data Field (may be omitted)

The length Indicator Field, Fixed Part and Variable Part constitute the Header of the TPDU.

8.2.1 Length Indicator Field

This field is contained in the first octet of the TPDUs. The length is indicated by a binary number, with a maximum value of 254 (11111110). The length indicated is the header length, including parameters, but excluding the length indicator field and user data, if any. The value 255 (11111111) is reserved for possible extensions.

8.2.2 Fixed Part

The fixed part contains frequently occurring functions including the code of the TPDU. The length and the structure of the fixed part are defined by the TPDU code, defined by bits 5 to 8 of the second octet of the header.

8.2.2.1 TPDU Code

This field contains the TPDU code and is contained in Octet 2 of the header. It is used to define the structure of the remaining header. This field is a full octet except in the following cases:

1110 xxxx	Connecting Request
1101 xxxx	Connection Confirm
0101 xxxx	Reject
0110 xxxx	Data Acknowledgement

Where xxxx (bits 4-1) is used to signal the CDT.

Any other bit pattern may be used to define a TPDU Code. Only those codes defined in Section 8.1 are currently valid.

8.2.3 Variable Part

The variable part is used to define parameters relating to optional functions. If the variable part is present, it shall contain one or more parameters. The number of parameters that may be contained in the varialbe part is indicated by the length of the variable part which is a LI minus the length of the fixed part.

Since the currently defined minimum fixed part for headers which allow parameters is four octets, and since the length indication field is limited to a maximum of 254, the maximum length of the variable part is 250 octets.

Each parameter contained within the variable part is coded as follows:

Octets	Bits 8 7 6 5 4 3 2 1
n+1	Parameter Code
n+2	Parameter Length Indication (e.g."m")
n+3	Parameter Value
n+2+m	

- o The parameter code field is coded in binary and, without extensions, provides a maximum number of 255 different parameters. However, as noted below, bits 8 and 7 indicates the source of definition, so the practical maximum number of different parameters is less. Parameter code 1111 1111 is reserved for possible extensions of the parameter code.

- o The parameter length indication indicates the length, in octets, of the parameter value field. The length is indicated by a binary number, "m" with a theoretical maximum value of 255. The practical maximum value of "m" is lower. For example, in the case of a single paramete r contained within the variable part, two octets are required for the Parameter Code and the Parameter Length Indication itself. Thus, the value of "m" is limited to 248. For larger fixed parts of the header and for each succeedimg parameter, the maximum value of "m" decrease s.

- o The parameter value field contains the value of the parameter identified in the parameter code field.

- o No standard parameter codes use bits 8 and 7 with the value 00.

- o Implementations shall accept the parameters defined in the variable part in any order. If any parameter is duplicated then the later value will be used.

8.2.3.1 Checksum Parameter (Class 4 only)

All TPDU types may contain a checksum parameter in their variable part. This parameter must always be present except when the non use of checksum option is selected.

Parameter Code: 1100 0011
Parameter Length: 2
Parameter Value: Result of checksum algorithm. This algorithm is specified in Section 6.18.

8.2.4 Data Field This field contains transparent user data. Restrictions on its size are noted for each command.

8.3 Connections Request (CR)

8.3.1 Structure

1	2		3	4	5	6	7	8	p	p+1
LI	CR	CDT	00000000	00000000	SOURCE-REFERENCE		class options	VARIABLE PART		USER DATA

8.3.2 LI

See Section 8.2.1

8.3.3 Fixed Part (Octets 2 to 7)

CR: Connection Request Code: 1110

CDT: Initial Credit Allocation (set to 0000 in Classes 0 and 1 when specified as preferred class).

SOURCE REFERENCE: Reference selected by the transport entity initiating the CR TPDU to identify the requested TC.

CLASSES: Bits 8-5 octer 7 defines the preferred Transport Protocol class to be operated over the requested TC. This field may take on one of the following values.

0000	Class 0
0001	Class 1
0010	Class 2
0011	Class 3
0100	Class 4

The CR contains the first choice of class in the fixed part as above. Second and subsequent choices are listed in the variable part if required.

Bits 4-1 of octet 7 are reserved for options to be used on the requested transport connection.

The use of bits 4-1 is as follows:

BIT	OPTION	
4	0	always
3	0	always
2	=0	use of normal formats
	=1	use of extended formats
1	=0	use of explicit flow control in Class 2
	=1	no use of explicit flow control in Class 2

Note:

1. It is not valid to request 'use of expedited data transfer' (Additional option parameter) and no use of explicit flow control in Class 2' (bit 1 = 1).

2. Bits 4 to 1 are always zero in Class 0 and have no meaning.

8.3.4 Variable Part (Octets 8 to p)

The following parameters are permitted in the variable part:

o Transport Service Access Point Identifier (TSAP-ID)

Parameter code 11000001 for the identifier of the Calling TSAP.

11000010 for the identifier of the Called TSAP.

If a TSPA-ID is given in the request it may be returned in the confirmation.

o TPDU size

This parameter defines the proposed maximum TPDU size (in octets including the header) to be used over the requested transport connection. The coding of this parameter is:

Parameter Code 11000000

Parameter value field

00001101 8192 octets (not allowed in Class 0 of 1)

00001100 4096 octets (not allowed in Class 0 of 1)

00001011 2048 octets

00001010 1024 octets

00001001 512 octets

00001000 256 octets

00000111 128 octets

Default value is 00000111 (128 octets)

Version Number (not used in Class 0)

Parameter code 11000100

Parameter value field 00000001

Default value 00000001

Default value 00000001 (not used in Class 0)

- o Security Parameter (not used in Class 0)

  This parameter is user defined.

  Parameter code 11000101

  Parameter value and length field are user defined

- o Checksum (not used in Classes 0 through 3)

  See Section 8.2.3.1

  This parameter must always be present in a CR TPDU requesting Class 4, even if the checksum selection parameter is used to request non-use of the checksum facility.

- o Additional Option Selection (not used in Class 0)

This parameter defines the selection to be made as to whether or not additional options are to be used.

Parameter code 11000110

Parameter length: 1
Parameter value field:

Bits related to options particular to one class are not meaningful and may take any value in the other classes.

BITS		OPTION
4	1=	Use of network expedited in Class 1
	0=	Non use of network expedited in Class 1
3	1=	Use of receipt confirmation in Class 1
	0=	Use of explicit AK variant in Class 1
2	0=	Checksums are to be used in Class 4
	1=	Checksums are not to be used in Class 4
1	1=	Use of transport expedited data transfer service
	0=	No use of transport expedited data transfer service

Default falue is 00000001

o Alternative protocol class (not used in Class 0)

Parameter code 11000111

Parameter length n

Parameter value encoded as a sequence of single octets. Each octet is encoded as for octet 7 but with bits 4-1 set to zero (i.e., no alternative option selections permitted).

o Acknowledge Time

This parameter conveys the maximum acknowledge time AL to the remote transport entity. It is an indication only, and is not subject to negotiation (see section 7.4.5.3).

Parameter Code 10000101

Parameter Value field: n a binary number (2 octets)

n is the maximum acknowledge time, expressed in milliseconds.

o Throughput   Parameter code: 10001001

Length : 12

	1st 3 octets :	Targer value, calling-called user direction
	2nd 3 octets :	Min. acceptable, calling-called user direction
	3rd 3 octets :	Target value, called-calling user direction
	4th 3 octets :	Min. acceptable, called-calling user direction

Values are expressed in octets per second.

o Residual error rate	Parameter code:	10000110
	Length :	3
	1st octet :	Target value, power of 10
	2nd octet :	Min. acceptable, power of 10
	3rd octet :	TSDU size of interest, expressed as a power of 2
o Priority	Parameter code:	10000111
	Length :	2
	Value :	Integer
o Transit delay	Parameter code:	10001000
	Length :	8
	1st 2 octets :	Target value, calling-called user direction
	2nd 2 octets :	Max. acceptable, calling-called user direction.

3rd 2 octets	:	Target value, called-calling user direction.
4th 2 octets	:	Max. acceptable, called-calling user direction

Values are expressed in milliseconds.

8.3.5 User Data (Octets p+1 to the end)

No user data are permitted in class 0, and are optional in the other classes. Where permitted, it may not exceed 32 octets.

8.4 Connection Confirm (CC)

8.4.1 Structure

1	2	3	4	5	6	7	8	p	p+1
LI	CC 1101	CDT	DST-REF	SOURCE-REF		class options	VARIABLE Part		USER DATA

8.4.2 LT

See Section 8.2.1.

8.4.3 Fixed Part (Octets 2 to 7)

CC	:	Connection Confirm Code: 1101
CDT	:	Initial Credit Allocation (set to 0000 in Classes 0 and 1).
DST-REFERENCE	:	Reference identifying the requested transport connection at the remote transport entity.
SOURCE REFERENCE	:	Reference selected by the transport entity initiating the CC TPDU to

identify the confirmed TC.

CLASSES : Defines the selected transport protocol class to be operated over the accepte
d

TC according to the negotiation rules specified in Section 6.5.

8.4.4 Variable part (Octet 8 to p)

See Section 8.3.4

8.4.5 User Data (Octets p+1 to the end)

See Section 8.3.5

8.5 Disconnect Request (DR)

8.5.1 Structure

LI	DR	DST-REF	SOURCE-REF	REASON	VARIABLE	USER DATA
	10000000				PART	

8.5.2 LI

See Section 8.2.1

8.5.3 Fixed Part (Octets 2 to 7)

DR : Disconnect Request Code: 1000

DST-REFERENCE : Reference identifying the TC at the remote transport entity.

SOURCE REFERENCE : Reference identifying the TC at the transport entity initiating the command. Value zero when reference is unassigned.

REASON : Defines the reason for disconnecting the TC. This field shall take one of the following values:

The following values can be used for class 1 to 4:

128 + 0 - Normal disconnect

initiated by session entity.

128 + 1 - Remote transport entity congestion at connect request time

*128 + 2 - Connection negotiation failed (i.e. proposed class(es) not supported).

128 + 3 - Duplicated connection detected

128 + 4 - Mismatched references

128 + 5 - Protocol error

128 + 6 - Not used

128 + 7 - Reference overflow

128 + 8 - Connection request refused on thi
s

network connection

128 + 9 - Not used

128 + 10 - Header or parameter length inval
id

The following values can be used for all classes.

0 - Reason not specified

1 - Congested at TSAP

*2 Session entity not attached to TSAP

*3 Address unknown

Note: Reasons marked with '*' may be reported to the TS-user as 'persistent', other reasons as 'transient'.

8.5.4 Variable Part (Octets 8 to 10)

o A parameter may be provided to allow additional information related to the clearing of the connection.

Parameter code: 11100000

Parameter Value Field: Additional information. This field is intended to be used by the transport service provider for internal purposes.

o Checksum (see 8.2.3.1)

8.5.5 User Data (Octets p+1 to the end)

Not allowed in class 0,

This field may not exceed 64 octers and is used to carry TS-User data. The successful transfer of this data is not guaranteed.

8.6 Disconnect Confirm (DC)

(Not used in Class 0)

8.6.1 Structure

```
1     2          3        4        5      6        7        p
LI               DST-REFERENCE  SOURCE-REFERENCE  Variable Part
   11000000
```

8.6.2 LI

See Section 8.2.1

8.6.3 Fixed Part (Octets 2 to 6)

DC : Disconnect Confirm Code: 1100

DST-REFERENCE : See Section 8.3.3

SOURCE-REFERENCE: See Section 8.4.3

8.6.4 Variable Part

Checksum (see 8.2.3.1)

8.7 Data (DT)

8.7.1 Structure

Normal Format for Class 0 to 1

```
1     2          3          4         5

LI    DT      E     TPDU-NR    User Data
   11110000 0
            T
```

Normal format for Class 2, 3 and 4

```
1        2        3        4          5          6       p     p+1
LI              DST-REFERENCE    E   TPDU-NR   Variable Part  User Data
   11110000                      O
                                 T
```

Extended Format for optional use in Classes 2,3 and 4

```
1       2      3       4     5,6,7,8    9        p p+1

LI   DT     DST-REFERENCE  E  TPDU-NR  Variable  User Data
   11110000                O
                           T
```

8.7.2 LI

Section 8.2.1

8.7.3 Fixed Part

(Classes 0 to 1 : - Octets 2 to 3; classes 2,3,4 normal format: Octets 2 to 5; classes 2,3,4 extended format: - Octets 2 to 8)

DT	:	Data Transfer Code: 1111
DST-REFERENCE	:	See Section 8.4.3
EOT	:	When set to ONE, indicates that the current DT TPDU is the last Data Unit of a complete DT TPDU sequence (End of TSDU).
TPDU-NR	:	TPDU Send Sequence Number (Zero in Class 0), may take any value in Class 2 without explicit flow control.

8.7.4 Variable Part

Checksum (See 8.2.3.1)

8.7.5 User Data Field

This field contains data of the TSDU being transmitted. The length of this field is limited to the negotiated TPDU size for this transport connection minus 3 octets in Classes 0 and 1, and minus 5 octets (normal header format) or 8 octets (extended header format) in the other classes. The variable part, if presemt, amy further reduce the size of the user data field.

8.8 Expedited Data (ED)

(Not used in Class 2 when "no explicit flow control" option is selected.)

8.8.1 Structure

Normal Format

1	2	3	4	EOT 5	6	p	p + 1
LI	ED 00010000	DST-REFERENCE		EDTPDU-NR 1.		Variable Part	User Data

Extended Format

1	2	3	4	EOT 5,6,7,8	9	p	p + 1
LI	ED 00010000	DST-REFERENCE		EDTPDU-NR 1.		Variable Part	User Data

8.8.2 LI

See Section 8.2.1

8.8.3 Fixed Part

(Octets 2 to 5, normal format: 2 to 8, extended format)

ED: Expedited Data command code: 0001

DST-REFERENCE: Same as Section 8.4.3

ED TPDU-NR: Expedited TPDU identification number (Classes 1, 3, and 4; may take any value in Class 2).

8.8.4 Variable Part

Checksum (See 8.2.3.1)

8.8.5 User Data Field

This field contains an expedited TSDU. Up to 16 octets.

8.9 Data Acknowledgement (AK)

Not applicable for Class 0 and Class 2 when the "no explicit flow control" option is selected, and for Class 1 when the network receipt confirmation option is selected.

Flow Control Confirmation (class 4 only - optionally used)

This parameter contains a copy of the information received in an AK TPDU, to allow the transmitter of the AK TPDU to be certain of the state of the receiving transport entity (See Section 7.4.5.6).

Parameter Code: 100001011

Parameter value field 64 bits, used as follows:

- Lower Window Edge (32 bits)
  Bit 32 is set to zero, bits 31 to 1 contain the YR-TU-NR value of the received AK TPDU. When normal format is in use, only the least significant seven bits (bits 1 to 7) of this field are significant.

- Your Sub-Sequence (16 bits)
  Contains the value of the sub-sequence parameter of the received AK TPDU, or zero if this parameter was not present.

- Your Credit (16 bits)
  Contains the value of the CDT field of the received AK TPDU. When normal format is in use, only the least significant four bits (bits 1 to 4) of this field are significant.

8.10 Expedited Data Acknowledgement (EA)

(Not applicable for Class 0 and Class 2 when the no explicit flow control option is selected).

8.10.1 Structure

Normal Format

1	2	3 4	5	6 p
LI	EA 00100000	DST-REFERENCE	. YR-TU-NR 0.	Variable Part

Extended Format

1	2	3 4	5,6,7,8	9 p
LI	EA 00100000	DST-REFERENCE	. YR-TU-NR 0.	Variable Part

8.9.1 Structure

Normal Format

1	2	3	4	5	6	p
LI	AK CDT 0110	DST-REFERENCE		. 0.	YR-TU-NR	Variable Part

Extended Format

1	2	3	4	5,6,7,8	9,10	11	p
LI	AK 01100000	DST-REFERENCE		. 0. YR-TU-NR	CDT		Variable Part

8.9.2 LI

See Section 8.2.1

8.9.3 Fixed Part

(Octets 2 to 5, normal format: 2 to 10, extended format)

AK: Acknowledgement command code: 0110

CDT: Credit Value (set to 0 in class 1)

DST-REFERENCE: Same as Section 8.4.3

YR-TU-NR: Sequence number indicating the next expected DT TPDU number.

8.9.4 Variable Part

Checksum (See 8.2.3.1)

Sub-sequence number (class 4 only - optionally used).

This parameter is used to ensure that AK TPDUs are processed in the correct sequence. If it is absent, this is equivalent to transmitting the parameter with a value of zero.

Parameter Code: 100001010

Parameter Value: 16-bit sub-sequence number.

8.10.2 LI

See Section 8.2.1

8.10.3 Fixed Part

(Octets 2 to 5, normal format; 2 to 8, extended format)

EA: Acknowledgement command code: 0010

DST-REFERENCE: Same as Section 8.4.3

YR-TU-NR: Identification of the ED TPDU being acknowledged. May take any value in Class 2.

8.10.4 Variable Part

Checksum (See 8.2.3.1)

8.11 Reject (RJ)

(Not used in Classes 0, 2, and 4)

8.11.1 Structure

Normal Format

1	2	3	4	EOT 5	6	p
LI	RJ 0101	CDT	DST-REFERENCE	0.	. YR-TU-NR	Variable Part

Extended Format

1	2	3	4	EOT 5,6,7,8	9,10	11	p
LI	RJ 01010000	DST-REFERENCE		. YR-TU-NR	CDT	Variable Part	

8.11.2 LI

See Section 8.2.1

8.11.3 Fixed Part

(Octets 2 to 5, normal format; 2 to 10, extended format)

RJ: Reject Command Code: 0101

CDT: Credit Value (set to 0 in class 1)

DST-REFERENCE: Same as Section 8.4.3

YR-TU-NR: Sequence number indicating the next expected TPDU from which retransmission should occur.

8.11.4 Variable Part

No parameters exclusive to this TPDU type.

8.12 TPDU Error (ERR)

1	2	3	4	5	6
LI	ERR 01110000	DST-REFERENCE		Reject Cause	Parameters

8.12.1 LI

See Section 8.2.1

8.12.2 Fixed Part

ERR: TPDU Error Code: 0111

DST-REFERENCE: Same as Section 8.4.3

REJECT CAUSE:

00000000 Reason not specified

00000001 Invalid parameter code

00000010 Invalid TPDU type

00000011 Invalid parameter value

8.12.3 Variable Part (Octets 6 to the end)

Parameter Code: 1100001

Parameter Value Field:

Contains the bit pattern of the rejected TPDU up to and including the octet which caused the rejection. This parameter is mandatory in Class 0.

Checksum (See Section 8.2.3.1)

SECTION THREE - CONFORMANCE

9. CONFORMANCE

Implementations claiming conformance to this standard shall:

1. Implement either Class 0 or Class 2 or both.

2. If other classes are implemented, the following rules shall be observed:

   a) If Class 3 or Class 4 is implemented then Class 2 must be implemented

   b) If Class 1 is implemented then Class 0 must be implemented.

3. The following table defines the requirements for the implementation of the options defined in previous sections:

	Class 0	1	2	3	4
TPDU with Checksum	no	no	no	no	m
TPDU without Checksum	m	m	m	m	o
Expedited Data Transfer	no	m	m	m	m
No Expedited Data Transfer	m	m	m	m	m
Flow Control in Class 2	no	no	m	no	no
No Flow Control in Class 2	no	no	o	no	no
7 bits format (normal)	m	m	m	m	m
31 bits format (extended)	no	no	o	o	o
Use of Receipt Confirmation in Class 1	no	o	no	no	no
No use of Receipt Confirmation in Class 1	no	m	no	no	no
Use of Network Expedited in Class 1, if T-EXPEDITED DATA necessary	no	o	no	no	no
No use of Network Expedited in Class 1, if T-EXPEDITED DATA necessary	no	m	no	no	no

o - optional: An implementation may or may not provide this user-selected option.

m - mandatory: An implementation must provide for this option

no - An implementation shall not provide this option.

4. Implementations claiming conformance shall support a TPDU length of 128 octets. If larger maximum TPDU

sizes can be supported in Classes 1,2,3, or 4, then all permitted TPDU sizes between the maximum and 128 octets shall be supported.

5. Claims of conformance shall state:

   a) which class of protocol is supported.

   b) which additional options indicated by the letter 'o' in the above table are supported.